ESTIMATION OF BIOMASS AT INNATE ERRORS OF METABOLISM INSTITUTE (IEIM)

CAMILO ENRIQUE MONCADA GUAYAZÁN

June 5th 2015

TABLE OF CONTENTS

List of Tables

Table of Figures

1. INTRODUCTION

This Project is a part of the main project named "*OPTIMIZACION DE LA PRODUCCION DE PROTEINAS RECOMBINANTES EN Pichia Pastoris BASADA EN UN MODELO EN SILICO"* developed altogether by the *Instituto de Errores Innatos del Metabolismo (IEIM)* and *Departamento de Electrónica* of the *PONTIFICIA UNIVERSIDAD JAVERIANA* which main objective is to improve the *Pichia Pastoris yeast* recombinants protein's production performance, since this yeast is highly employed for this purpose because of its capability to produce different proteins by modifying its DNA and producing a specific strain of the microorganism. To achieve proteins production is required a controlled environment for the yeast inside a Bioreactor to first grow and be nurtured *(Growth Phase)* increasing its *Biomass (quantity of microorganism inside the Bioreactor)* and lastly by changing the *Substrate* or feed from Glycerol to Methanol, being the latter the agent by which the yeast reacts and starts to produce the protein.

It is necessary to make clear that to keep the bioreactor a controlled setting it is full automated via Labview using different measurements and controllers for this objective. Nevertheless, such important variables as *Biomass (X) & Substrate (S)* aren't available immediately; they are acquired through chemical laboratory procedures that take time and manual work by the institute technicians and bacteriologists. For this reason, it became indispensable to make available these variables or states by some mechanism or procedure in order to have a better grasp and knowledge of the process. It was selected to do the previous for the *Growth Phase* only, since in the institute there is no mechanism to distinguish at what moment change the substrate and begin with the *production phase.*

The latter generated the goal for this project which is by means of the measurements of the process and other tools, algorithms, etc. and obtain the estimate of the *Biomass* at the current time. This project presents the design and development of the estimator that fulfill this necessity first with a model based estimator that involves the identification of the system's model differential equations *(Nonlinear)*, then using this model and incorporated to a Kalman estimation filter making the estimator more robust and reliable against noises and uncertainties *(Sections 3 & 4.1-4.3).* Therefore, continue with the alternative of a soft or virtual sensor, which is a data driven or direct estimator. This sensor also utilizes the data or measurements to do the estimates via the direct filtering, unlike the model based estimator skips the model identifying of the process, here was done by developing a *Neural Network*. The next step is to select the appropriate estimator for implementation purposes and finally achieve the estimator of the *Biomass* in Labview environment. Here all the development, considerations, design process and results are presented through the document.

1.1. General Objective

Design and implement an online Biomass estimator for the *Pichia Pastoris* yeast bioreactor at the Errores Innatos del Metabolismo Institute (IEIM).

1.1.1 Specific Objectives

- Develop a growth model phase model for *Pichia Pastoris* bioreactor at the Errores Innatos del Metabolismo Institute (IEIM).
- Desing and implement in MATLAB Extended Kalman and Unscented Kalman observers for the developed Model.
- Design and implement in MATLAB a virtual sensor (Soft Sensor) data driven black box based on the experimental data of IEIM's bioreactor.
- Evaluate with experimental data of IEIM's bioreactor the estimators and select the one with best performance between Kalman observer and virtual sensor.
- Implement the selected estimator in the bioreactor supervisory and control system's at IEIM.

2. BIOREACTOR

The harvest and production of the *pichia pastoris* yeast is done in a bioreactor *(ref. KLF 200 by Bioengineering)* controlled environment. The process takes two a stage approach, the first one is the growth phase, where initial amounts of the yeast is put through a batch process with Glycerol as substrate/food, with the sole purpose of increasing the quantity of Biomass inside the bioreactor and obtain the maximum quantity to start with the final stage. This phase objective is to maximize the production of the protein as was done in (1) although the substrate is switched to a continuously controlled feed of Methanol, since the *Pichia Pastoris* is genetically altered to react to this particular substrate and produce a specific protein according to the modified strain of the microorganism.

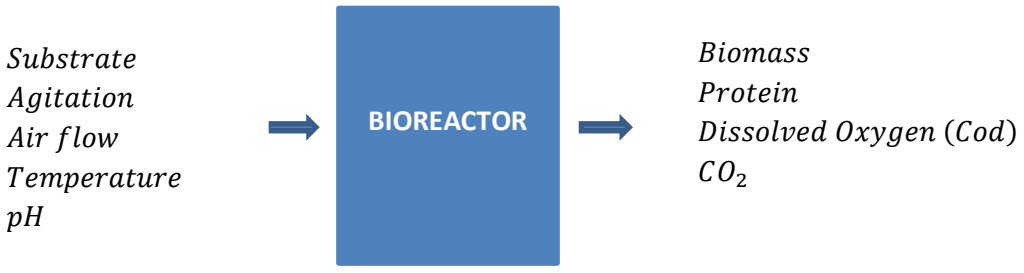

Figure 2-1. Bioreactor Process

Figure 2-1 describes the bioreactor process. The controlled variables are the quantity of substrate for the respective phase, air flow and agitation to regulate the transfer of oxygen into the medium and finally the temperature and pH, since it's a living organism it affects is metabolism and more importantly its survival conditions. The outputs of the process are the desired outcome that in the end is protein production, but it's also required to know Biomass dynamics and value and confirm the appropriate evolution of the process which for now is done but the perish and experience of the process operator by assessing the historic data of the measured and controlled variables during the experiment, which we can add the Dissolved Oxygen Concentration *(Cod)* and carbon Dioxide(CO_2). The Bioreactor and process characteristics (2) are enlisted next:

- Bioreactor: Bioengineering KFL2000 3.7 Liters.
- Yeast Volume: 1.6 – 2,0 Liters.
- Operation Temperature: 28°C +/- 0.1°C.
- Growth Phase duration: 12-24 hours.
- Protein production time: 6-7 days.
- Oxygen probe response time: 80 seconds.
- Other Measurements Acquisition response time: every 10 seconds.
- Agitation: 0-1500 RPM.

The main variables used during the description of this project are presented in table Table 1:

Variable	*Symbol*	*Units*	*Description*
Biomass	$X(t)$	$\frac{g_{Bio}}{l}$	*Concentration of Biomass present in the bioreactor in grams per litre of solution*
Glycerol	$S(t)$	$\frac{g_{Gly}}{l}$	*Concentration of Glycerol present in the bioreactor in grams per litre of solution*
Dissolved Oxygen Concentration	$Cod(t)$	*adimensional* (%)	*Concentration of* ***Dissolved Oxygen*** *present in the bioreactor in grams per litre of solution*
Agitation	$rpm(t)$	*RPM*	*External input to manipulate oxygen's transference rate, pH and medium's temperature*

Table 1. Bioreactor's Variables.

As explained before the goal of this project is to estimate the biomass present inside the bioreactor at the current time throughout the whole process, although for academic and consistency purposes Substrate it's estimated as well. The approach used is describe in Figure 2-2, where a software based estimator is created to do such task. For the estimator to work, regardless of its architecture, it is needed the initial conditions of the variables *Biomass (X)*, *Glycerol (S)* and *Dissolved Oxygen Concentration (COD)* and the measurements of *Dissolved Oxygen Concentration* $(\widetilde{Cod}(t))$ during the course of the experiment, which is the only variable that can be acquired instantaneously during the process. As for the model Based estimator it is also necessary the input control value of *Agitation*($\widetilde{rpm}$).

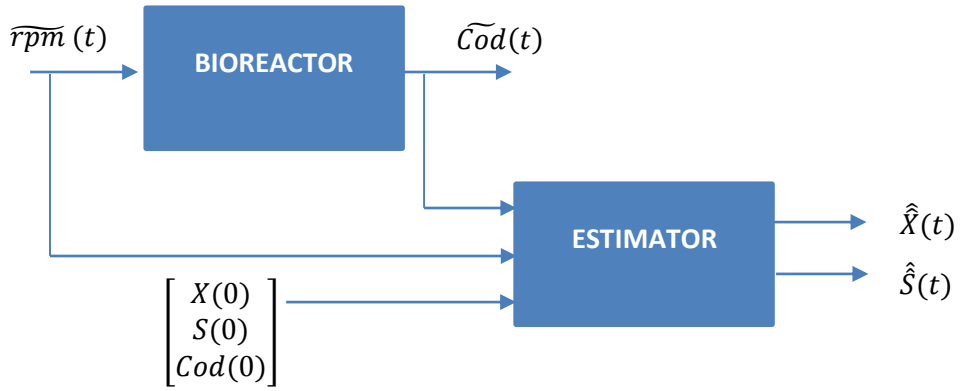

Figure 2-2. Estimator Architecture.

3. BIOREACTOR MODEL

The first of the two approaches used to estimate the quantity of biomass in the bioreactor in this work are model based. The model is described by an ordinary differential equations set, so the first step to system identification by this way is to have the set of data or experiments for training and validation purposes, which is introduced next.

3.1. Experiments

The data set initially collected for this work was extracted from bioreactor shown in 2 starting with six different experiments of the same *Pichia Pastoris* yeast. The diversity lies in the initial condition for *Glycerol* for the growth phase and the *Methanol* set point for the production phase among all the set.

The Data for *Biomass (X)* and *Glycerol (S)* were acquired manually on offline laboratory and chemical tests from samples taken mostly every two hours, which provide results later during the course and after the end of the experiment, this sample times will be called as t_X . Meanwhile, *Dissolved Oxygen Concentration (COD)* and the input *agitation (RPM)* are obtained online during the run of the test via the automation system (LabView) every 10 seconds. Due to end application and result purposes the data for *COD* and *RPM* are averaged for a time period of fifteen minutes (15') and it's named as t_{Cod}, since this is an appropriate time to estimate the biomass according to the process times and dynamics, and this average also works as a pre stage filtering for the data variations of the sets.

The units for *Biomass* and *Glycerol* are displayed in percentage value of *grams per litre (g/l)* for both, simple percentage for *Dissolved Oxygen Concentration (COD)* and RPM's for *agitation*.

The experiments were analyzed in order to select which ones will be suitable for either training or validation of the estimator. The features to distinguish the sets between approved or discarded are data dynamics, outliers/errors and consistency. Two examples of the aforementioned analysis are shown in Figure 3-1 and Figure 3-2, displaying examples of an accepted and a rejected experiment respectively.

*A remark to be noted from here on throughout the lenght of the document is that the experimental data will be denoted in red color with asterisks (i.e. *****), while simulated data will be denoted with blue color either using a line for continuous data or a blue cross for discrete simulated results (i.e. **–** or **+**).*

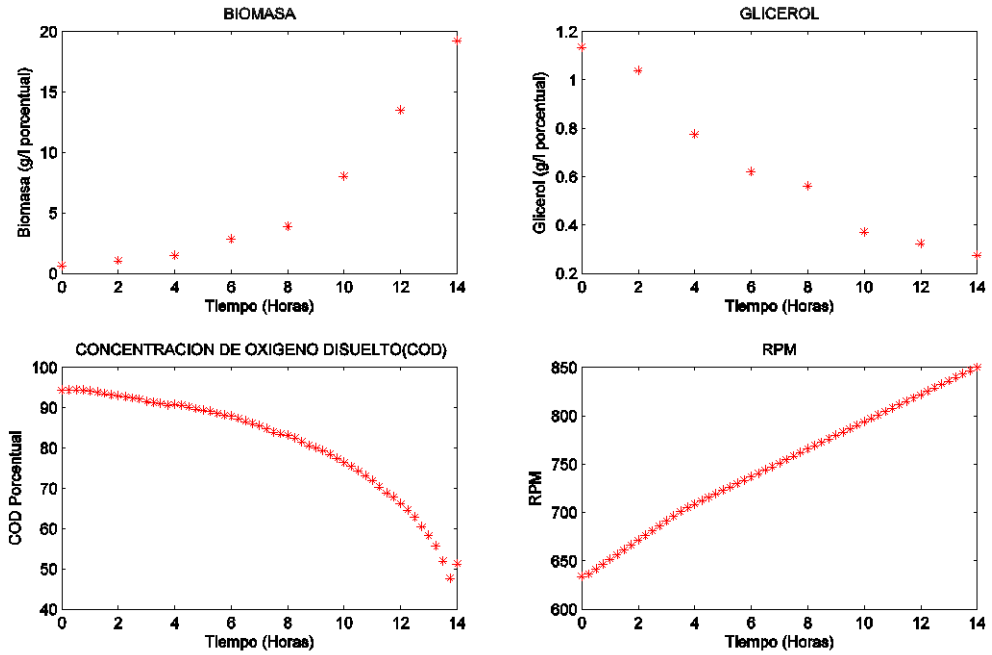

Figure 3-1. Experiment 1% Glycerol 0.1% Methanol data (exp11).

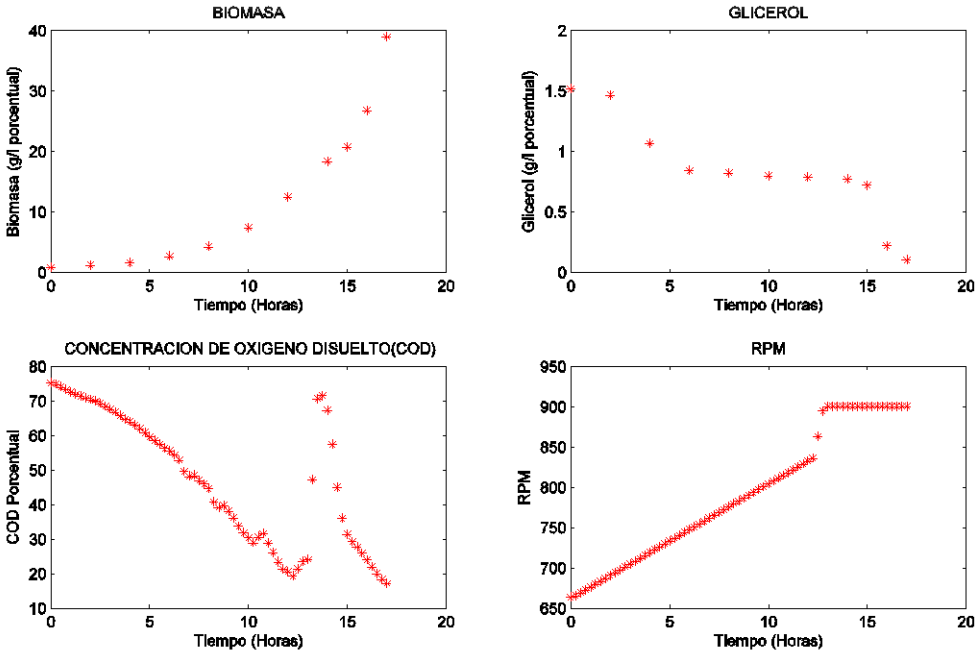

Figure 3-2. 2% Glycerol - 0.5% Methanol experimental data (exp25).

Figure 3-1 displays the data for experiment denoted as *exp11*. The data is very consistent with a biological process where the *Biomass* growth is related to the quantity of *Glycerol* available and also to the consumed *COD*. This experiment is considered as good data for training or validation, although de *Glycerol* value at 12th hour was linearly interpolated because this value was higher than the previous one, which is wrong in the sense that a substrate is always being consumed and can only decrease during the course of the run. For this reasons the experiment is accepted for use.

Figure 3-2 shows the data for *exp25*. Observing the data, it is clear that during the time span from the 6th hour to the 15th hour the Glycerol only decrease in a 0.13% *g/l* while the Biomass keeps growing during the same interval; from 2.62 to 20.66% *g/l*; which is a complete inconsistency. For this reason this experimental data is discarded.

A general remark for all the experimental data regarding the agitation input (RPM's), its values were from specific profile; which could be a type of ramp, constant or staircase signal; for the given experiment.

The initial conditions for each of the experiments evaluated for use are shown in:

Experiment % Glycerol % % methanol duration

Experiment	$\%Glycerol[g/l]$	$\%Methanol[g/l]$	*Duration* [*hours*]
exp11	1 %	1 %	14
exp15	1 %	5 %	16
exp21	2 %	1 %	18
exp25	2 %	5 %	17
exp41	4 %	1 %	25
exp45	4 %	5 %	26

Table 2. Experiment Characteristics.

Finally, the experiments that are retained and accepted for the system's training/validation are: *exp11, exp15, exp21* and *exp45.*

3.2. Growth Phase Model

Now that the experimental data has been selected, the next step is to determine what model or models will be tested or identified according to their differential equations. First, it is necessary to clarify that the model is only for the growth phase of the process, when the yeast is being fed with *Glycerol* as substrate. The model for the production phase can be found at (1), the model obtained in this project complements and completes the whole process for the yeast.

After reviewing the literature [(3) (4) (5) (6) (7)] the first model proposed for identification is:

$$\begin{aligned} \dot{X} &= \mu(S) * X \\ \dot{S} &= r_{gly} * \mu(S) * X \\ \dot{Cod} &= a * rpm + b(100\% - Cod) + r_{Cod} * \mu(S) * X \end{aligned} \qquad \text{Eq. 3-1}$$

Where:

$$\mu(S) = \mu_{max} * \frac{S}{S + K_s}$$

μ_{max}: $Maximum\ growth\ rate$ [adimensional]

K_s: $Mid\ point\ Glycerol\ Saturation\ for\ Biomass$ [g_{Gly}/l]

r_{gly}: $Glycerol\ consumption\ rate$ $[\frac{g_{Gly}/l}{g_{Bio}/l}]$

a: $Agitation\ transference\ rate$ [%Cod/rpm]

b: $Saturation\ transference\ rate$ [adimensional]

r_{Cod}: $Oxygen\ consumption\ rate$ $[\frac{\%Cod}{g_{Bio}/l}]$

Where $\mu_{max}, K_s, r_{gly}, a, b \; and \; r_{cod}$ are the parameters of the model to be found, and rpm is an external control variable of the process for the agitation at the current time. Therefore, is worth to and pH by automatic controls to maintain the adequate conditions for the yeast to grow.mention that there are other variables being regulated at the process like Temperature, air flow,etc.

The core of the model is the *Monod Kinetic Model* - $\mu(S)$ (MKM) which outlines the growth rate for a bioreactor or bioprocess. MKM is a monotonously increasing function that simulates the evolution of the system, in which the *Biomass* will continue to grow as long as there is substrate (Nutrient); for the *Pichia Pastoris* yeast is *Glycerol* (S). The first two differential equations define this dynamics; observe how the two states mentioned before are linked in their dynamics as is shown below:

$$\dot{X} = \mu_{max} * \frac{S}{S + K_s} * X$$

$$\dot{S} = r_{gly} * \mu_{max} * \frac{S}{S + K_s} * X$$

One alternative for system's model proposed in this project, the Glycerol differential equation has been modified hence that it is saturated by the *COD* differential state. Accordingly, with this variation the second model to be identified in this project is shown as follows:

$$\dot{X} = \mu_{max} * \frac{S}{S + K_s} * X$$

$$\dot{S} = r_{gly} * \mu_{max} * \frac{S}{S + K_s} * \frac{Cod}{Cod + K_G} * X$$

$$\dot{Cod} = a * rpm + b(100 - Cod) + r_{Cod} * \mu_{max} * \frac{S}{S + K_s} * X$$

K_G: $Mid\ point\ Cod\ Saturation\ for\ Glycerol$ [%Cod]

$*\ The\ other\ parameters\ are\ the\ same\ as\ before.$

3.3. Identification Problem Formulation

Using the model defined in the section before as example, we proceed to define the identification problem as a minimization one as follows:

$$min_{\mu_{max}, K_s, r_{gly}, a, b, r_{Cod}} \sum_{i=1}^{N} [Q_{11}(i) * \left(\hat{X}_i - \tilde{X}_i\right)^2 + Q_{22}(i) * \left(\hat{S}_i - \tilde{S}_i\right)^2$$

$$+ Q_{33}(i) * \left(\widehat{Cod}_i - \widetilde{Cod}_i\right)^2]$$

$$s.t.$$

$$\dot{X} = \mu(\hat{S}) * \hat{X}$$

$$\dot{S} = r_{gly} * \mu_{max} * \frac{\hat{S}}{\hat{S} + K_s} * \hat{X}$$

$$\dot{Cod} = a * rpm + b(100 - \widehat{Cod}) - r_{Cod} * \mu(\hat{S}) * \hat{X}$$

$$\mu(\hat{S}) = \mu_{max} * \frac{\hat{S}}{\hat{S} + K_s}$$

$$rpm = \widehat{rpm}$$

$$min_{\mu_{max}} < \mu_{max} < MAX_{\mu_{max}}$$

$$min_{K_s} < K_s < MAX_{K_s}$$

$$min_{r_{gly}} < r_{gly} < MAX_{r_{gly}}$$

$$min_a < a < MAX_a$$

$$min_b < b < MAX_b$$

$$min_{r_{Cod}} < r_{Cod} < MAX_{r_{Cod}}$$

$Q(t)$: $simulation\ error\ weighting\ diagonal\ Matrix.$

The objective of the problem is to adjust the parameters in order to minimize the quadratic error between the simulated ($\hat{X}_i, \hat{S}_i\ and \widehat{Cod}_i$) and measured states($\tilde{X}_i, \tilde{S}_i\ and \widetilde{Cod}_i$). This problem can be solved for each one of the experiments data set as singles, or one whole problem for all the experiments used for training identification, finding the best model parameters set that simulate the dynamic of the bioreactor from the specified model and minimize the simulation error. In this project the problem was setup so that experiments *exp11, exp15 and exp21* were used for the identification phase and *exp45 as* validation set.

It has to be noted that the optimization problem is nonlinear due to that both models have nonlinear differential equations. The problem can be setup by the Matrix *Q,* which weights the significance given to the each state error in the objective function, subsequently the range of magnitudes of the states is different. *Biomass* range is within 0 to 80 maximum, *COD* is 40 to 90 and *Glycerol span is [0, 4].* The initial Q matrix weights for the problem was selected by using the $1/(Max_{state})^2$ criteria.

Also the parameters constraints for the intervals ($[min_{par}, MAX_{par}]$) forms the optimization problem setting the feasible area of search for each parameter. Because of the nonlinearity and high

complexity of the problem, these intervals should be nearby to the optimum solution because the initial conditions of the optimization problem establish the solution's development towards a local optima or divergent values and not a global optima. The initial values for the intervals were selected from REF DANI.

One issue for the problem's formulation is that because the data was sampled at different frequencies; every two hours for the *Biomass*/*Glycerol* and for *COD/RPM* at a fifteen minute rate (after averaging the 10" samples); generating time points where the data is not available for the objective function for the first two appointed variables. The solution for this matter is to define a time variant *Q* matrix, where it will have full rank diagonal with all its weighting values during the times where the *Biomass* and *Glycerol* exists (t_X) otherwise it will have zeros for this corresponding weights, the latter is expressed as follows:

$$Q(i) = \begin{cases} \begin{bmatrix} Q_X & 0 & 0 \\ 0 & Q_S & 0 \\ 0 & 0 & Q_{COD} \end{bmatrix} & ,i = f(t = t_X) \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & Q_{COD} \end{bmatrix} & ,i = f(t \neq t_X) \end{cases} \qquad 9$$

The Q matrix; respective values $[Q_{X,}\ Q_S, Q_{COD}] = 10^{-2}[2.52, 15.87, 7.043]$ and parameter intervals were tuned during the identification process for better results.

3.4. Results and Validation

The system was identified using *ACADO Toolkit* for Matlab (8). During the development of the tests it was determined that *exp45* distorted or biased the results of the identification, adjusting the model to minimize the error presented in this experiment disrupted too much the other experiments of the validation set, for this cause *exp45* was discarded as an outlier data.

Given the small set of experimental data approved (only three experimental sets), in order to maximize the efficiency of data for its information value, all the models were initially identified using each of the experiments as the validation set and then analyze which one provides the best performance. The different results for all the proposed models are shown in the next tables below.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
exp11 – val	4.885	16.34	0.4211	0.047	14.1244	90.44
exp15	1.377	1.28	0.183	0.0126	11.353	39.463
exp21	0.742	0.337	0.0126	0.039	13.872	129.55

Table 3.Staged Monod Model using exp11 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
exp11	1.152	0.594	0.272	0.027	4.726	12.161
exp15 – val	1.335	1.107	0.257	0.018	15.834	47.838
exp21	1.04	0.727	0.378	0.37	11.321	129.55

Table 4.Staged Monod Model using exp15 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
exp11	2.308	1.823	0.202	0.03	5.993	21.154
exp15	1.746	2.078	0.133	0.0105	5.443	23.671
exp21 – val	10.75	60.477	0.332	0.114	18.212	290.66

Table 5. Staged Monod Model using exp21 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
exp11 – val	5.261	19.382	0.404	0.043	26.092	218.965
exp15	2.257	5.649	0.169	0.0094	5.546	23.637
exp21	4.281	13.488	0.202	0.036	7.859	59.382

Table 6. Standard Monod Model using exp11 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
exp11	1.63	1.6802	0..191	0.033	2.431	5.905
exp15 – val	1.36	2.015	0.112	0.0095	12.478	45.473
exp21	3.28	0.11.16	0.264	0.068	6.517	42.293

Table 7. Standard Monod Model using exp15 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
exp11	1.936	1.006	0.198	0.037	5.663	17.791
exp15	1.3001	1.3	0.115	0.012	5.791	25.998
exp21 – val	2.499	6.585	0.285	0.0605	14.921	66.032

Table 8. Standard Monod Model using exp21 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
$exp11-val$	5.113	18.383	0.388	0.04	12.448	64.662
$exp15$	3.949	10.519	0.274	0.023	4.857	21.062
$exp21$	3.032	8.283	0.224	0.029	7.68	56.796

Table 9. Saturated Monod Model using exp11 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
$exp11$	2.708	6.159	0.227	0.038	1.6254	1.346
$exp15-val$	2.836	7.962	0.13	0.019	14.803	51.594
$exp21$	4.62	20.211	0.27	0.073	6.512	42.267

Table 10. Saturated Monod Model using exp15 as validation data.

Experiment	X		S		Cod	
	RMSE	Var	RMSE	Var	RMSE	Var
$exp11$	3.375	11.22	1.139	0.827	37.364	376.121
$exp15$	7.544	54.811	1.629	2.984	29.396	339.105
$exp21-val$	3.48	16.449	0.285	0.087	15.994	118.047

Table 11. Saturated Monod Model using exp21 as validation data.

After reviewing the different performances of the data set combinations, the best performance is achieved by the set where the *exp15* is used as the validation data. Then, we use this configuration as the standard one to train and validate the models and estimators used throughout this work.

After that last remark, the first Model identified is the standard Monod Model, its results with the given data set are showed next:

3.4.1. Monod Standard

As can be observed the standard Monod model has the differential equations for the *Biomass* and *Glycerol* decoupled from the *COD* variable. This allows to identify the system by stages, first the parameters for the differentials equations of *X* and *S* states, and then continue with the *COD* equation. This methodology is the first approach for the identification of this particular model, the identification results are shown next:

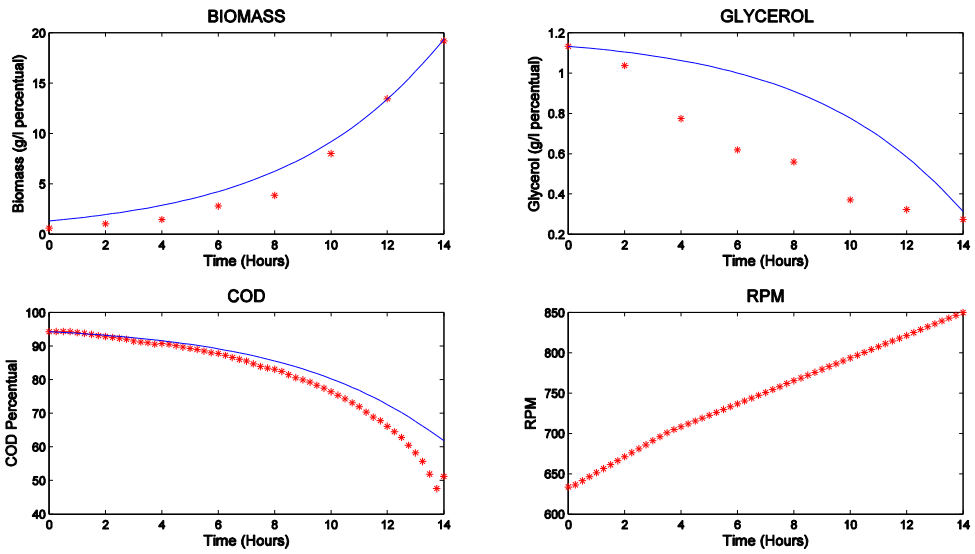

Figure 3-3. Exp11 staged Monod identification.

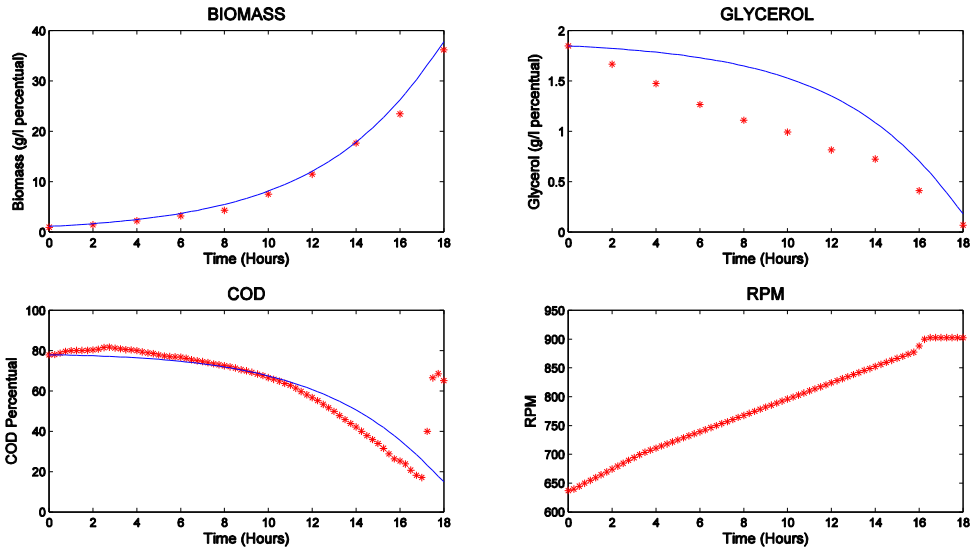

Figure 3-4. Exp21 staged Monod identification..

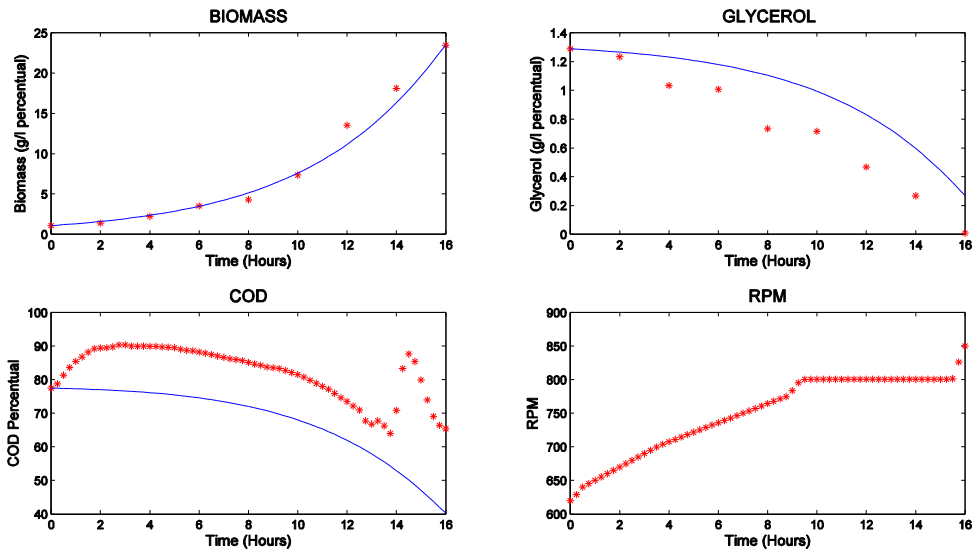

Figure 3-5. Exp15 Staged Monod Validation.

Figure 3-3 and Figure 3-4 are the results on the training data sets and were identified as a multiple experiment optimization problem. All the states were estimated with some congruency, but Glycerol has the tendency to be above the measurements over the length of the experiments. While Figure 3-5 shows how the parameters identified with the previous experiments perform on the validation data *exp15*, here the Biomass and Glycerol accomplish the same tendencies and errors, nonetheless for *COD* it is not the same since it underestimates presenting a higher error than in the training data.

The parameters, Root Mean Squared error (RMS) and variances for all the models obtained will be summarized and compared at the end of the section.

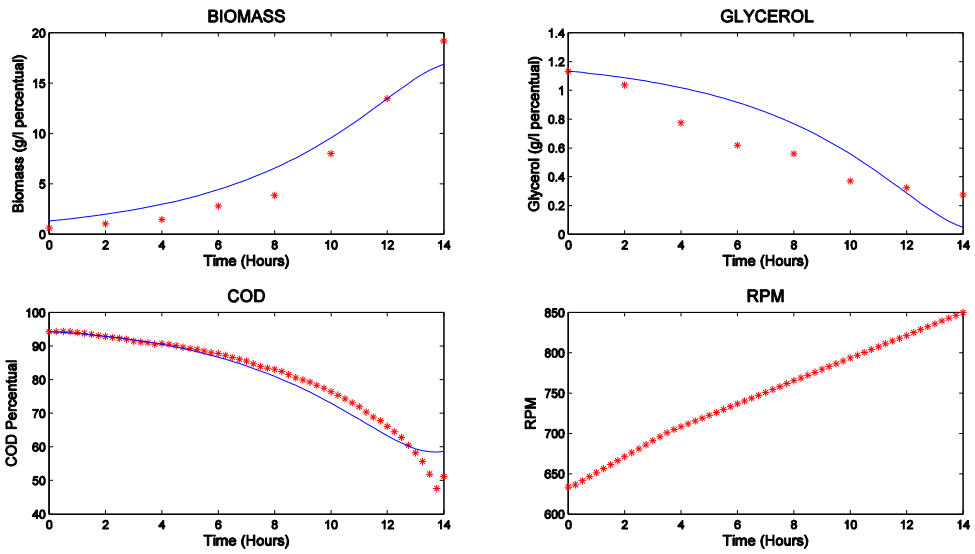

Figure 3-6. Exp11 Monod all parameters.

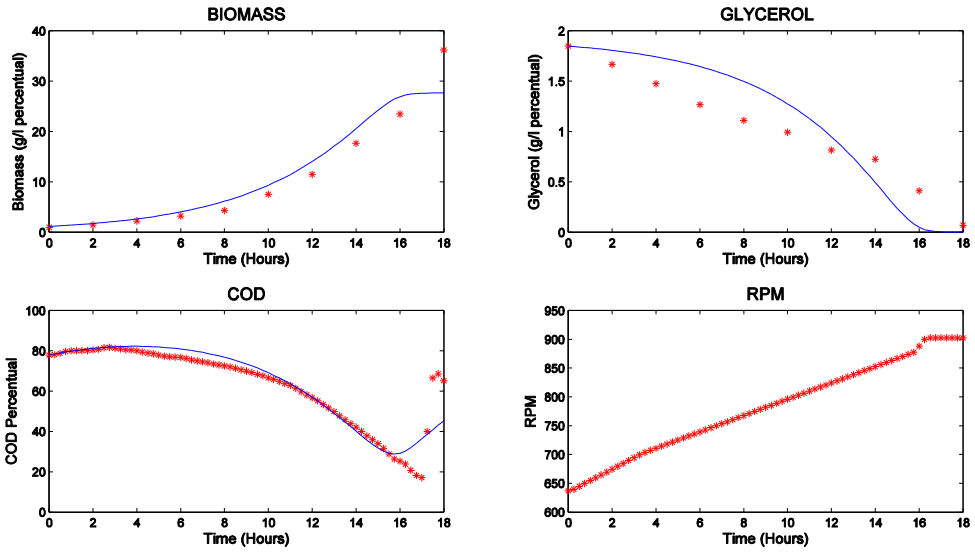

Figure 3-7. Exp21 Monod all parameters.

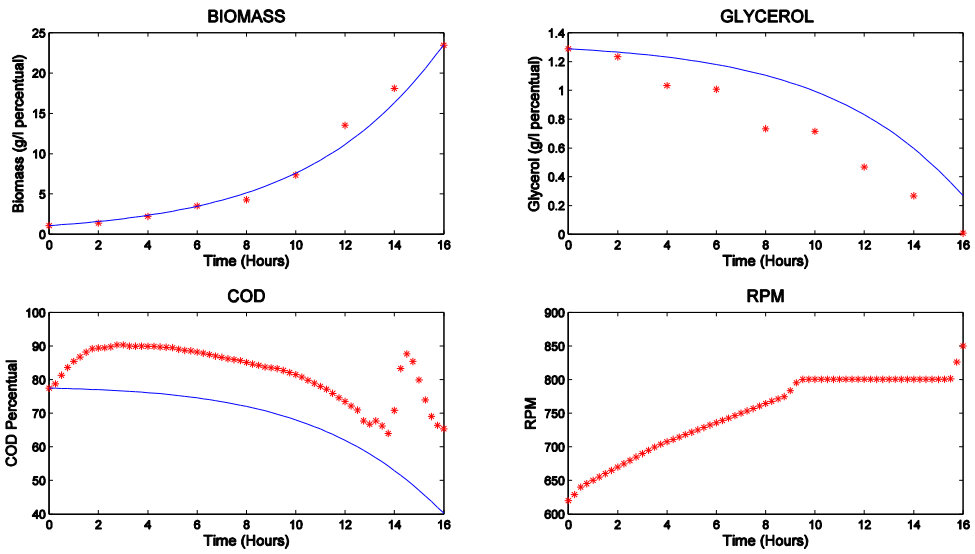

Figure 3-8. Exp15 validation Monod All parameters.

Equally as the Staged Monod the model presents good performance for all the variables, but the dynamics are different since Biomass presents an end time saturation, Glycerol the same and also this time does not keep over the data but goes under at the end and gets to zero value with consumption dynamics. As for *COD* also at the end of the experiment the parameters allow the state to go upwards again as the data shows. All of these tendencies are shown also in the validation set.

3.4.2. Substrate Oxygen Saturated

This Model was defined in Eq. 3-2. The identification results are shown below:

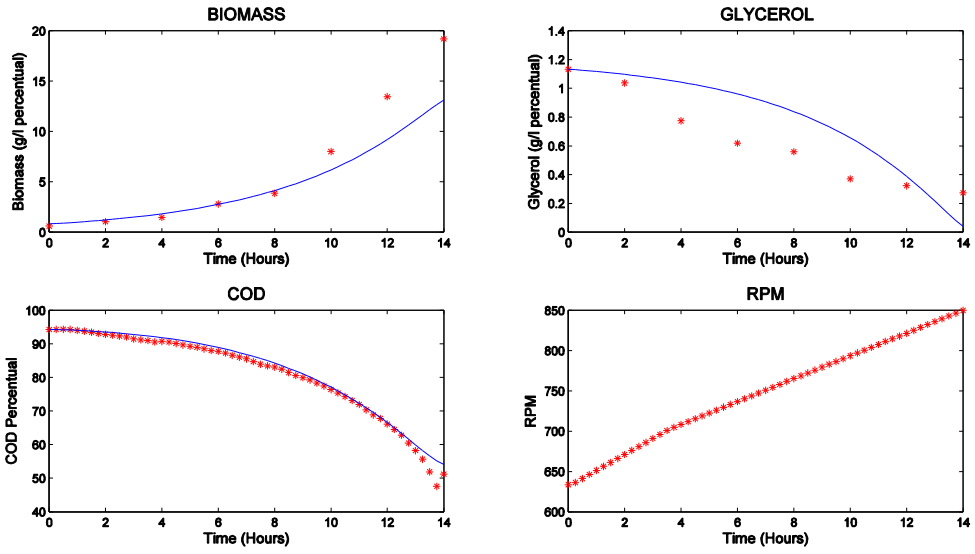

Figure 3-9. Exp11 Monod Saturation.

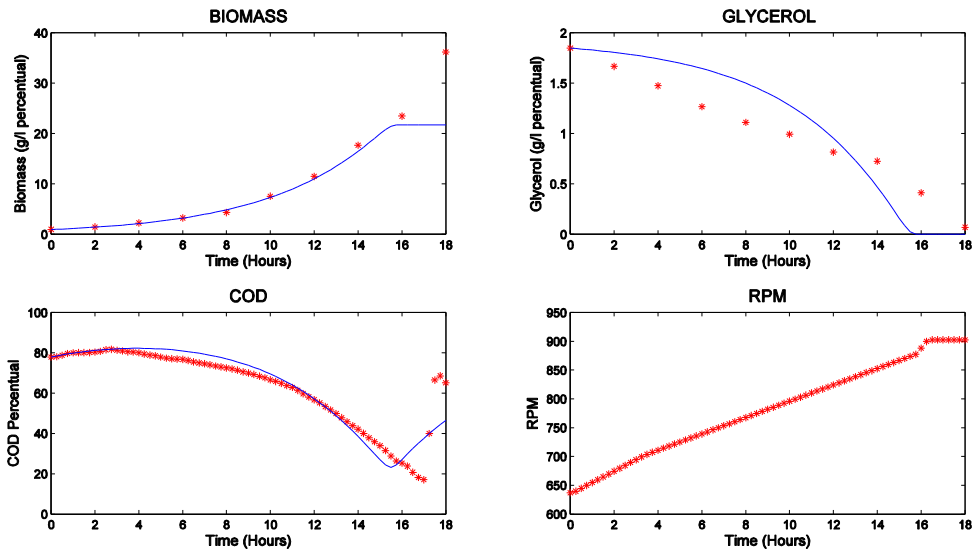

Figure 3-10. Exp21 Monod Saturation.

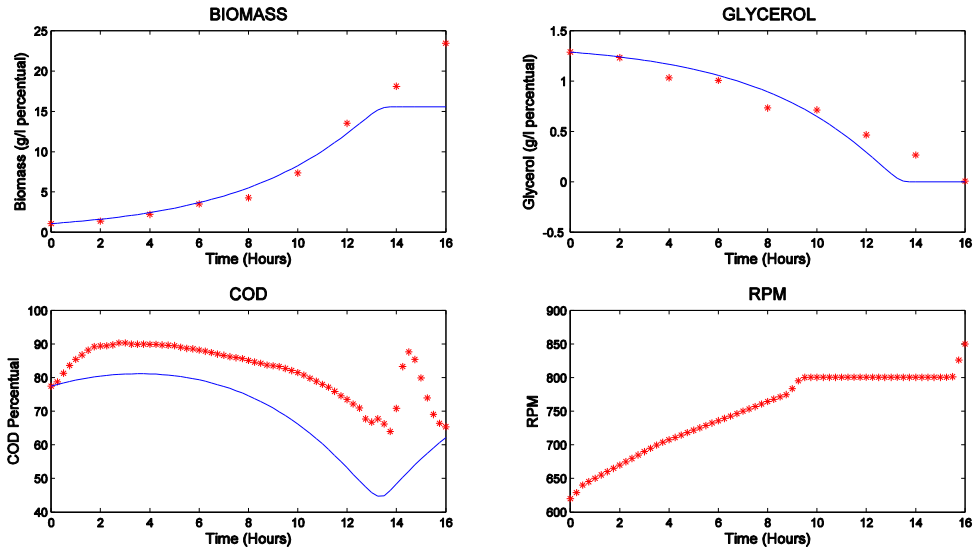

Figure 3-11. Exp15 Monod Saturation Validation.

During the different tests it was obvious that the K_g parameter always converged to the minimum value of the interval, what this states is that the parameter has no influence on the model and tends to be nullified. The dynamics of the model are very similar to the previous Monod, with the remark that this saturates faster towards the end for all the states.

The parameters of all the models are shown in Table 12:

Parameter	*Staged Monod*	*Monod Complete*	*Glycerol Saturated*
μ_{max} $[\boldsymbol{units}]$	$2.028 * 10^{-1}$	$2.292 * 10^{-1}$	$2.098 * 10^{-1}$
$\boldsymbol{K_s}$ $[\boldsymbol{units}]$	$4.593 * 10^{-2}$	$1.417 * 10^{-2}$	$2.978 * 10^{-2}$
$\boldsymbol{r_{gly}}$ $[\boldsymbol{units}]$	$-9.2146 * 10^{-3}$	$-1.598 * 10^{-2}$	$-1.864 * 10^{-2}$
$\boldsymbol{a}$ $[\boldsymbol{units}]$	$9.999 * 10^{-7}$	$1.599 * 10^{-6}$	$1.16 * 10^{-7}$
$\boldsymbol{b}$ $[\boldsymbol{units}]$	$1.292 * 10^{-2}$	$1.552 * 10^{-1}$	$1.551 * 10^{-2}$
$\boldsymbol{r_{cod}}$ $[\boldsymbol{units}]$	1.956	5.008	6.36
$\boldsymbol{K_g}$ $[\boldsymbol{units}]$	**	**	$1 * 10^{-6}$

Table 12. Model's Paramaters.

** $-K_g$ *parameter doesn't exist for this models.*

Observing Table 12, parameters $\mu_{max}, K_s, r_{gly}, a \text{ and } b$ are around a similar order for the all the different models displaying some consistency, although they produce diverse dynamics. Factor $\boldsymbol{a}$ $(agitation\ transference\ rate)$ always tended to minimum interval value or close to this limit, oxygen transference to the medium isn't that much for any rpm value. Meanwhile, $\boldsymbol{r_{cod}}$ highlights for its wide range of values, none is close to another between the models. $\boldsymbol{r_{cod}}$ magnitude order is in $(1x10^0)$, probing there's high oxygen consumption during the growth phase of the yeast

We define the RMS error for each variable as the following:

$$RMS_{ERROR} = \sqrt{\frac{1}{N}\sum\left(\hat{X}_i - \tilde{X}_i\right)^2} = [RMSE_X, RMSE_S, RMSE_{COD}]$$

States	*Staged Monod*		*Monod Complete*		*Glycerol Saturated*	
	RMSE	*Var*	*RMSE*	*Var*	*RMSE*	*Var*
$\boldsymbol{X}$ $[\boldsymbol{g/l}]$	1.335	1.107	1.36	2.015	2.836	7.962
$\boldsymbol{S}$ $[\boldsymbol{g/l}]$	0.257	0.018	0.112	0.0095	0.13	0.0186
$\boldsymbol{Cod}$ [%]	15.834	47.838	12.478	45.473	14.803	51.594

Table 13. RMS error and Variances for Identified Models.

Table 13 presents the Root Mean Squared Error and Variances for the validation experiment (*exp15*), which in this case is *exp15*, for all the states accordingly. The most difficult state to estimate was the *Cod* given its changing and nonlinear dynamics. For *Biomass* and *Glycerol* the estimations were in acceptable values.

4. STATE ESTIMATION

Now that the model has been tuned, the next step for model based estimation is to incorporate the model into the Kalman filter and build a robust estimator against noise and uncertainty. First it is reviewed the Kalman's algorithm for a linear system and modify it to a nonlinear system.

4.1. Kalman Filter

The objective of the Kalman filter/observer is to estimate all the system's states based on the space states model and the measurement of one or more outputs (9). As said before given the discrete space states model with uncertainties defined as:

$$X_{k+1} = A * X_k + B * U_k + w \;\; ; \;\; Y_k = C * X_k + v \; ;$$
$$Q_p \;\&\; R_m : Process\ and\ measurements\ noises$$
$$/uncertainties\ covariances\ matrices\ \ for\ w\ and\ v\ respectively$$

The Kalman filter used in this project has the following steps to make the prediction of the states recursively:

$$- Prediction\ Step -$$
$$X_{k+1}^{-} = A * X_k^{+} + B * U_k$$
$$P_{k+1}^{-} = A * P_k^{+} * A^T + Q_p$$
Escriba aquí la ecuación.
$$- Correction\ Step -$$
$$E = Y^{k+1} - C * X_{k+1}^{-}$$
$$S = C * P_{k+1}^{-} * C^T + R_m$$
$$K = P_{k+1}^{-} * C * S^{-1}$$
$$X_{k+1}^{+} = X_{k+1}^{-} + K * E$$
$$P_{k+1}^{+} = P_{k+1}^{-} - K * C * {P_{k+1}^{-}}^T$$

Where $Q_p \; and \; R_m$ are tunable parameters for the filter weighing the importance given to the model dynamics and the measurement directly taken from the plant respectively. The other factor required to initialize the filter are the initial conditions of the states.

The models identified in *BIOREACTOR MODEL* chapter area clearly nonlinear and for this reason it's necessary to make some adjustments to the Kalman filter algorithm, such alternatives for this project are Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) for nonlinear systems, the architecture for both Kalman filters are exactly as presented in Figure 2-2.

4.2. Extended Kalman Estimator (EKF)

The modifications made to the Kalman filter (linear version) to adapt it for nonlinear systems are to linearize the nonlinear model equations $f(X_k^+, u_k)$ obtained in section 3 via Jacobian and apply the algorithm as was defined before. But for a more precise estimation in the prediction for the *a priori* state the nonlinear system equations are used. These modifications are shown next:

$$- Prediction\ Step -$$

$$A_k = \left.\frac{\partial f(X,u)}{\partial X}\right|_{X=X_k^+}$$
$$X_{k+1}^- = X_k^+ + \Delta t * f(X_k^+, u_k)$$
$$P_{k+1}^- = A_k * P_k^+ * A_k^T + Q_p$$

This is how was implemented the EKF algorithm in Matlab, the modification are only at the prediction step because the measurement modeling is still lineal. Then proceed with the filtering of the identified models in 3.4.

4.2.1. Monod Standard EKF

First is the Monod Standard of 6 parameters, the results of the Extended Kalman Filter for this model are shown below:

The best $Q_p\ and\ R_m$ matrices that model respectively the process disturbances (for each states regarding $X, S\ \&\ Cod$) and measurement noise for this model are:

$$Q_p = 1 * 10^{-7} * \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 100000000 \end{bmatrix}$$

$$R_m = [7 * 10^2]$$

As for the results of the simulations:

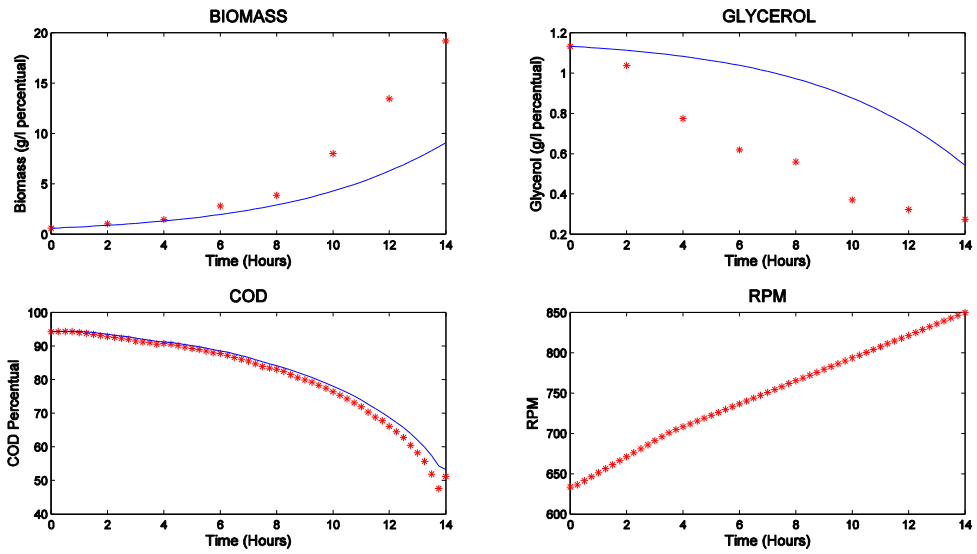

Figure 4-1. EKF for exp11.

This experiment shows a reduced performance in both variables *X* and *S*, increasing the estimation error, but improves state *COD* as a result of acquiring the measurement of this variable.

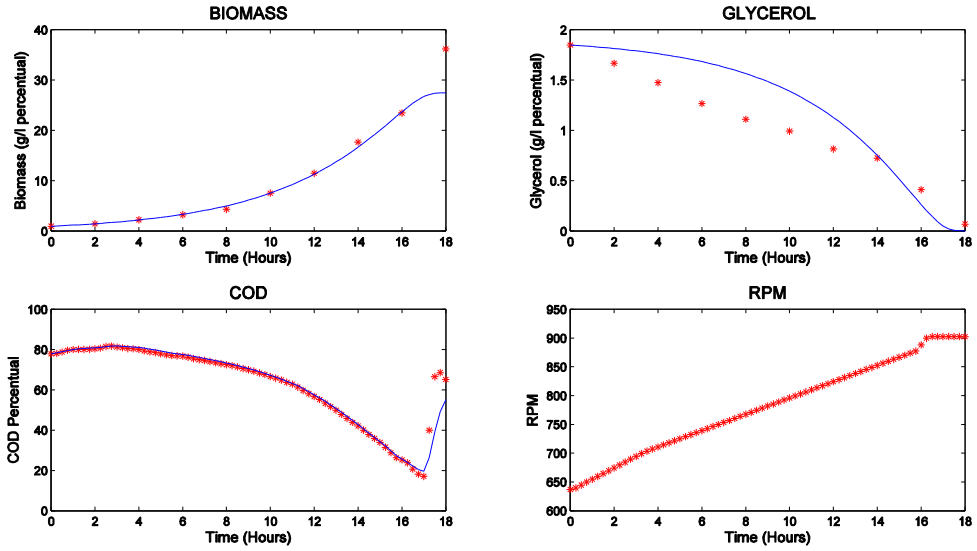

Figure 4-2. EKF for exp21.

For *exp21* the performance of the filter compared to the model alone is very similar and equally to experiment 11, while *COD* state improves with the measurement.

The validation was done with *exp15* likewise as for the identification process.

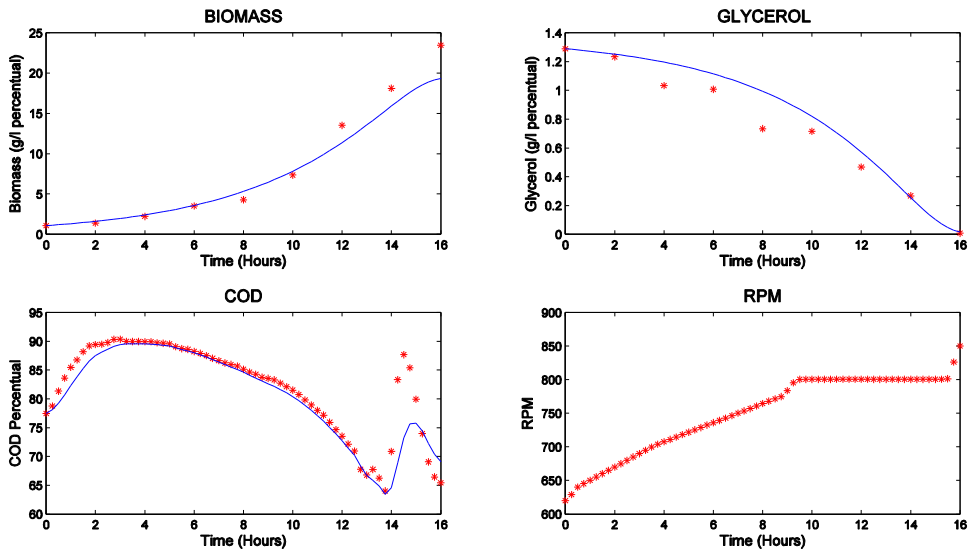

Figure 4-3. Validation data of the EKF with exp15.

The validation results are very good, the *COD* improves and *Biomass* and *Glycerol* keep their dynamics compared to the model simulation.

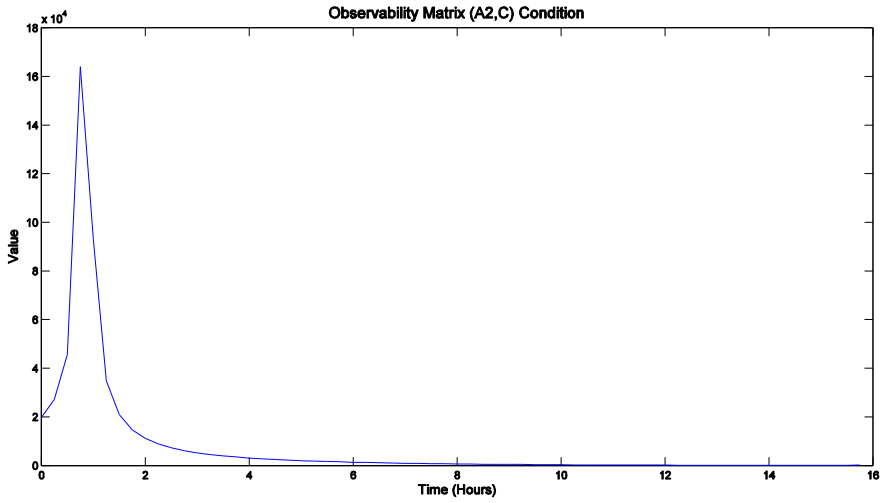

Figure 4-4. Observability condition for the system.

Since we have to discretize and linearize the System to obtain Matrix A_k, it is necessary to verify if the pair (A_k, C) is observable by checking the norm 2 condition number of the observability matrix $O = obsv(A_k, C)$ as can be seen in Figure 4-4 , although it has a non-observability peak around hour 1, but consistently decreases through time keeping the system detectable for all the states.

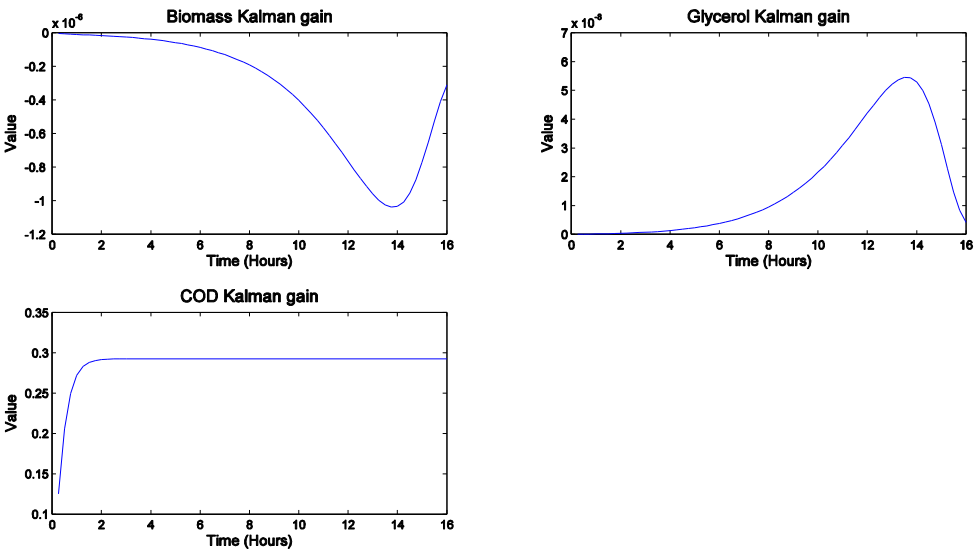

Figure 4-5. Kalman filter Gains for Validation experiment.

The Kalman gains ($K = P_{k+1}^{-} * C * S^{-1}$) are shown in Figure 4-5, they either are converging to zero or steady in a constant value, proving that the filter is stable.

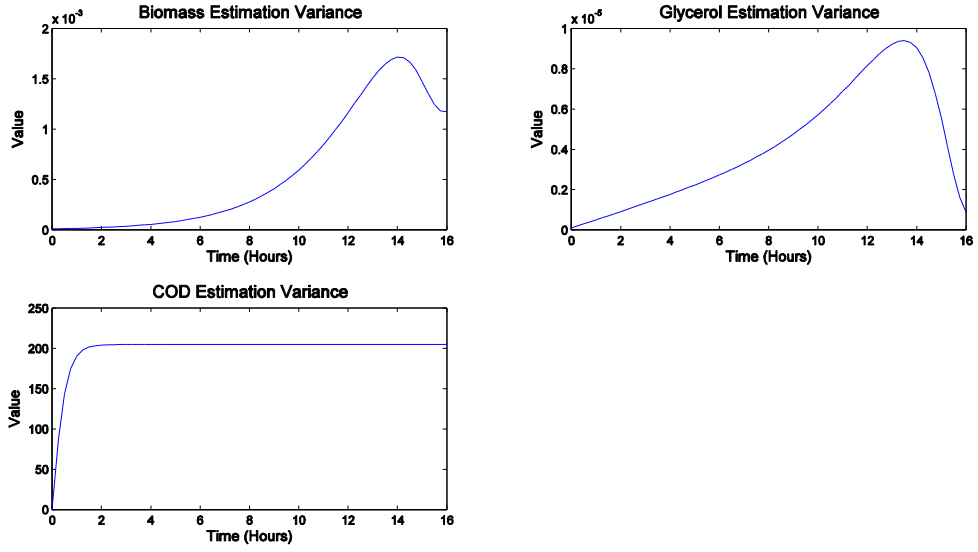

Figure 4-6. EKF Validation variances.

Equally the filter variances for each state are presented in Figure 4-6, and confirm how the estimation variance for each state are stable and converge either to zero or a constant value. Showing the consistency and stability of the estimation.

4.2.2. Substrate Oxygen Saturation Model EKF

Now we proceed with *Substrate saturated model* identified in 3.4.2 and its results are shown next:

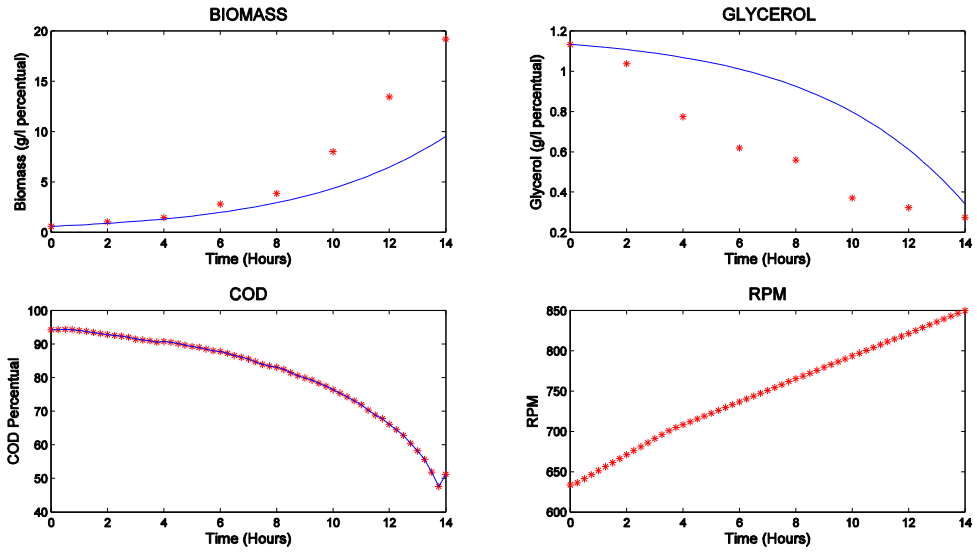

Figure 4-7. EKF for exp11 with Saturation Model.

In Figure 4-7 can be observed once more how the *Biomass* presents a dynamic under the experimental data. Contrary, *Glycerol* is over the real values for the entire run. These exhibits have a higher error than in the model simulation. As for *Cod* the estimation presents zero error because the filter takes the measurement as the estimate, this comes from the optimum filter parameters.

Figure 4-8 shows *exp21* with better simulation results than *exp11* for *Biomass* and *Glycerol* up until the 16th hour, but for the last two hours both states presents instability and ripple on their values. These behaviors were existent for different simulation parameters and hard to eliminate, the results shown are when these dynamics were minimized.

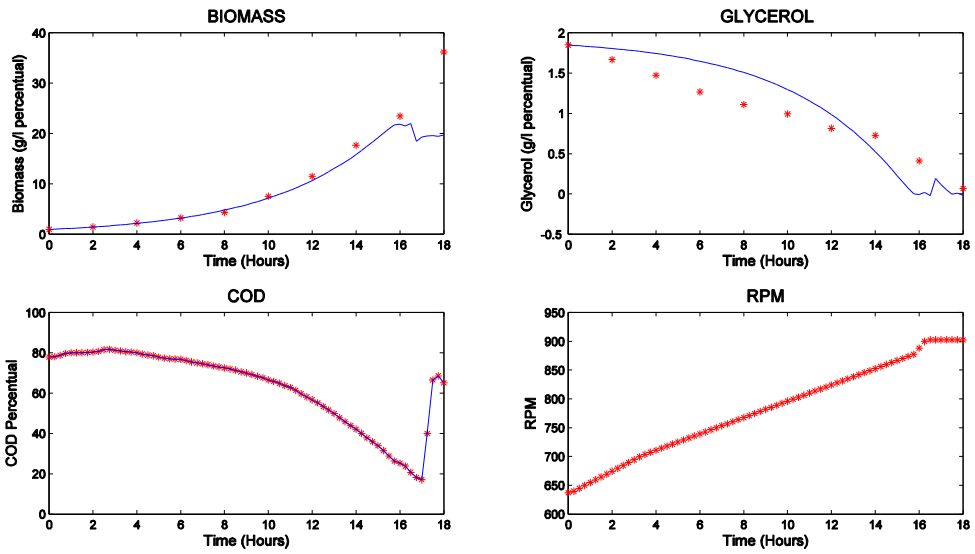

Figure 4-8. EKF for exp21 with Saturation Model.

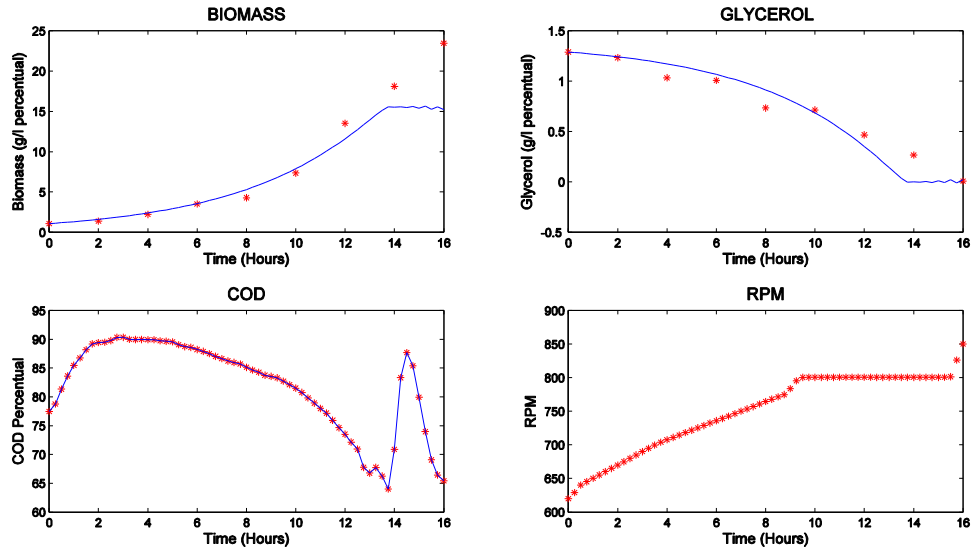

Figure 4-9. EKF for validation with Saturation Model.

Results of validation experiment *(exp15)* shown in Figure 4-9, displays an end time saturation with a small ripple similarly to *exp21*. Featuring and confirming how this filter tends to produce erroneous estimations for *Biomass* and *Glycerol*. *Cod* state was estimated perfectly because the filter practically takes the measurement and takes it as true value.

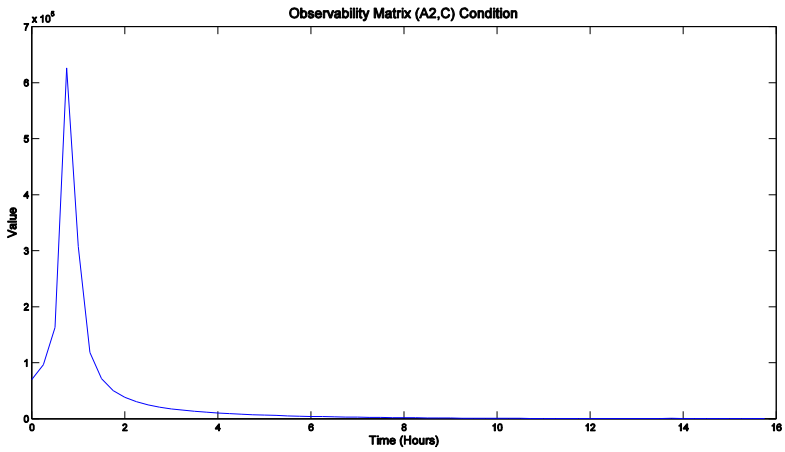

Figure 4-10. Observability condition for EKF Saturation.

Figure 4-10 exhibits the observability condition for this model, in the same way as the previous filter. Once more it has a peak close to the first hour *(1h)*, however the order of this peak value is one order greater 10^5 compared to the 10^4, disturbing the covariance matrix P_{k+1}^{-} during this time. Similarly the observability converges to zero towards the end of the experiment.

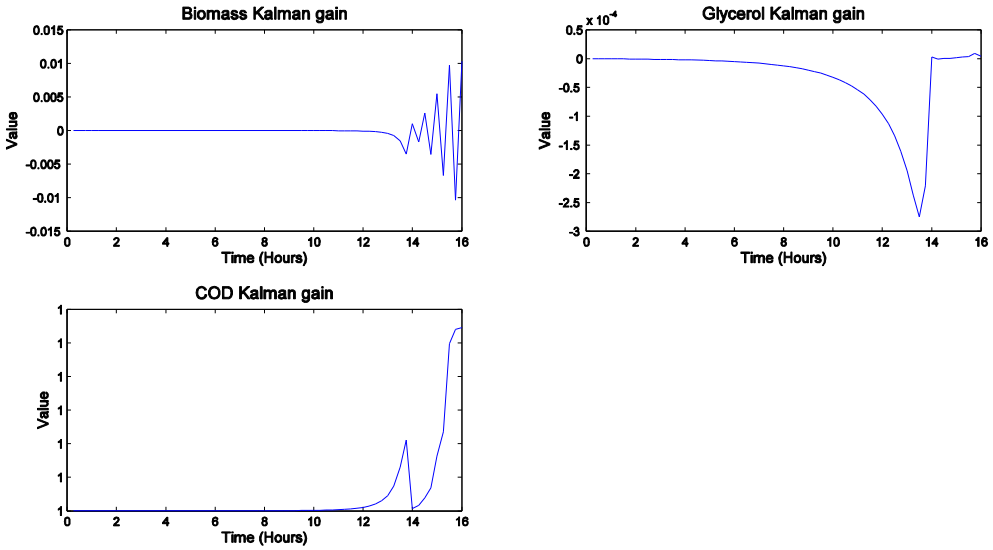

Figure 4-11. Kalman gains for each State, EKF saturation.

Figure 4-11 illustrates the instability of the Kalman Gains filter for the last two hours for the *Biomass* state. The latter confirms why there's some ripple for all the states at the end of the simulation. While *S* and *Cod* exposed some disturbances but finally converge to a constant value.

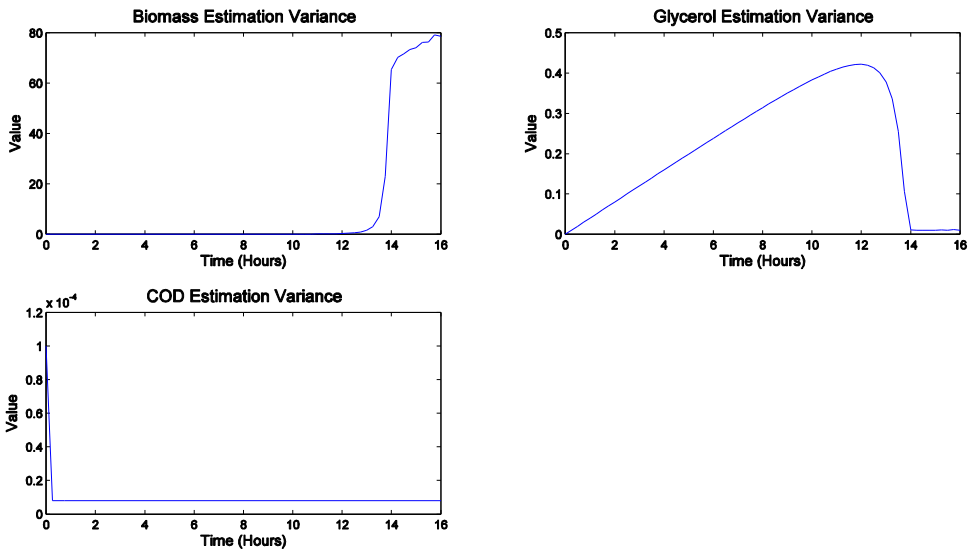

Figure 4-12. Variances for each state EKF Saturation.

Regarding the variances shown in Figure 4-12, all over again *Biomass* displays a concern at the last two hours, ratifying the issues with the estimation of this state, since its variance keeps increasing in this interval and never steadies at a constant value.

Known all these performance characteristics from the *Saturated Model EKF* it can be concluded that this filter isn't feasible and useful for *Biomass* estimation, which is the final objective of the project. Regardless of this remark below are the performance features for both filters:

States	*Standard Monod EKF*		*Substrate Saturated EKF*	
	RMSE	*Var*	*RMSE*	*Var*
$\boldsymbol{X}$ $[\boldsymbol{g/l}]$	1.755	2.885	2.9737	8.3551
$\boldsymbol{S}$ $[\boldsymbol{g/l}]$	0.119	0.0081	0.1245	0.0174
$\boldsymbol{Cod}$ [%]	3.255	8.9	0.0000	0.000

Table 14. EKF's RMS error and Variances for all states.

As Table 14 displays, the best EKF for state estimation is the one with Standard Monod Model. The *Biomass* and *Glycerol* estimation for this model is better for this states of both filters. While *Cod* estimation improves with the measurement, compared to the model's simulation in 3.4.1. Whereas *X* and *S* regress a little bit against those simulations.

4.3. Unscented Kalman Estimator

The alternative for the model based Kalman Filter is using the Unscented Transformation *(UT)* to approximate a probability distribution after the nonlinear transformation employing a set of points *(sigma points)* (10) (11). The goal is to utilize the UT to estimate the Covariance matrix P_k^- in the prediction step of the Kalman Filter.

First let us define the sigma points for the UT as follows:

$$-Sigma\ points-$$
$$x_k^{(0)} = X_k^+$$
$$x_k^{(i)} = X_k^+ \pm \sqrt{c * P_k^+}$$

L Sigma points are created around the value to be calculated according to its covariance. Then, they are evaluated through the nonlinear function to estimate the transformed mean and covariance, as indicated next:

$$-Nonlinear\ Mean\ \&\ Covariance-$$

$$y_{k+1}^{(i)} = X_k^+ + \Delta t * f(X_k^+, u_k)$$
$$\mu = \sum W_m^{(i)} * y_{k+1}^{(i)}$$
$$\Sigma = \sum W_c^{(i)} \left(y_{k+1}^{(i)} - \mu\right) * \left(y_{k+1}^{(i)} - \mu\right)^{\mathrm{T}}$$

The Nonlinear points are pondered when calculating the mean and covariance, those weighs are defined as:

$$W_m^{(0)} = \frac{\lambda}{c}$$
$$W_m^{(i)} = \frac{1}{2c}$$
$$W_c^{(i)} = W_m^{(i)}$$
$$W_c^{(0)} = \frac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta)$$

The parameters for the Unscented Transformation are:

$$-Parameters-$$

$$L : Number\ of\ States$$
$$\alpha, \beta\ \&\ k_i : Tunnable$$
$$\lambda = \alpha^2(L + k_i) - L$$
$$c = L + \lambda$$

Thus, the prediction step for the Unscented Kalman Filter *(UKF)* is established as:

$$- Prediction -$$
$$X_{k+1}^{-} = X_k^{+} + \Delta t * f(X_k, u_k)$$
$$P_{k+1}^{-} = \Sigma(X_k^{+}, P_k^{+}, W_m, W_c)$$

The correction step for the filter remains the same as for the EKF. The results of the UKF are shown in the next sections.

4.3.1. Monod Standard UKF

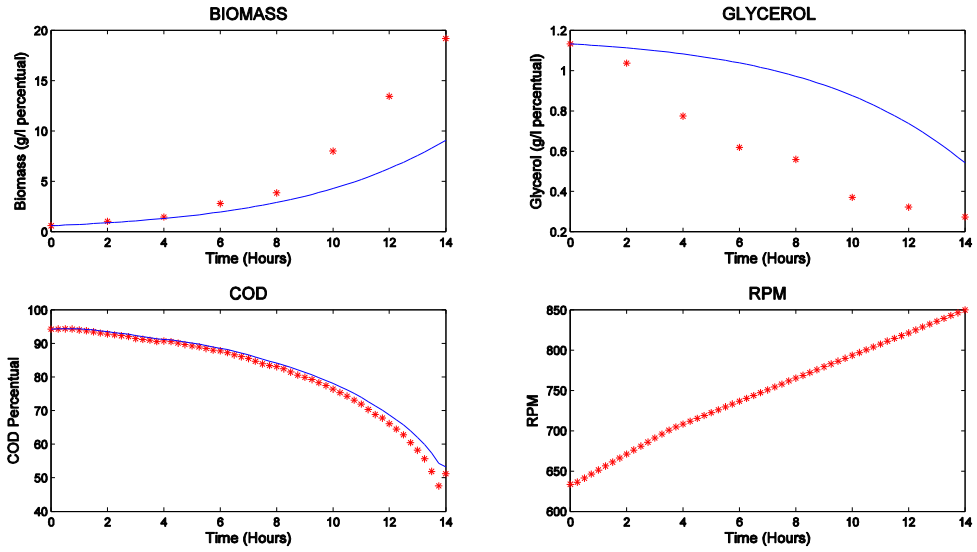

Figure 4-13. UKF for exp11 Standard Monod Model.

Results for *exp11* are shown in Figure 4-13. *Biomass* and *Glycerol* are poorly estimated since are they are under and over the measurements respectively. Presenting high error over the last five hours for *X* state and for *S* state error starts from the second hour.

Figure 4-14 displays the simulation results of the UKF for *exp21*. The states are estimated better compared to *exp11*; which tend to follow real data; except for the last two hours for *Biomass*, while *Glycerol* trails well the last four hours.

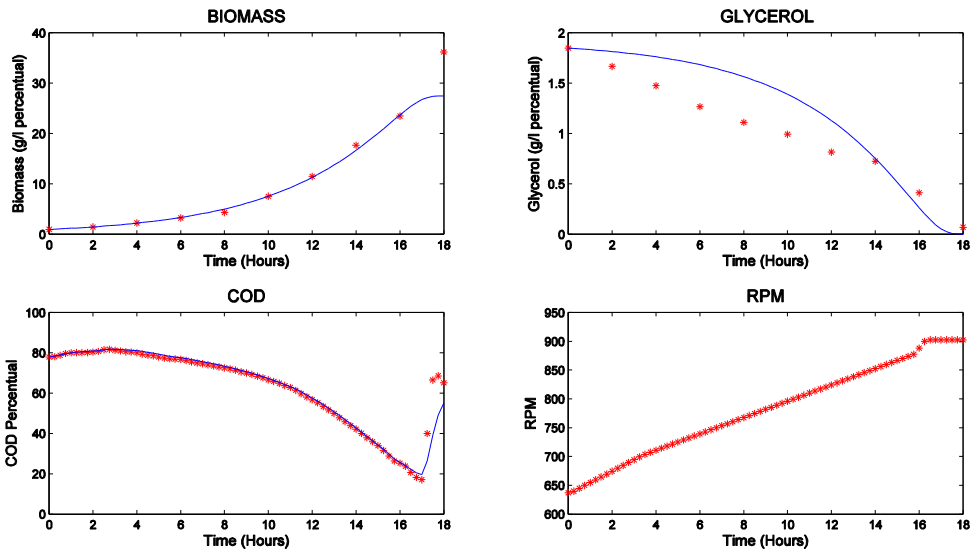

Figure 4-14. UKF for exp21 Standard Monod Model.

Repeatedly *exp15* was used for validation, its outcome is displayed next:

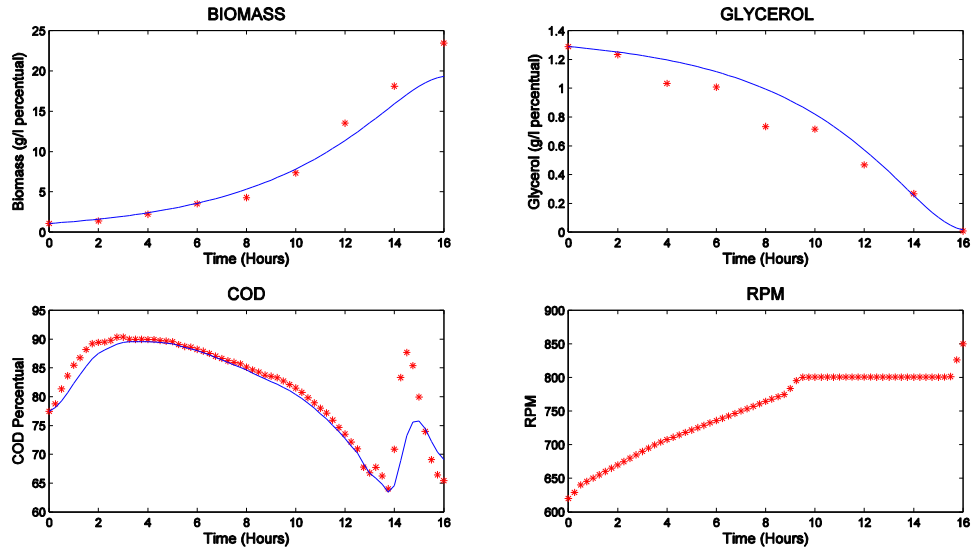

Figure 4-15. UKF validation with exp15 for Standard Monod Model.

Figure 4-15 shows really good simulation results for the unmeasured states. It tracks the system dynamics closely demonstrating a small error. *Glycerol's* end value is reached perfectly.

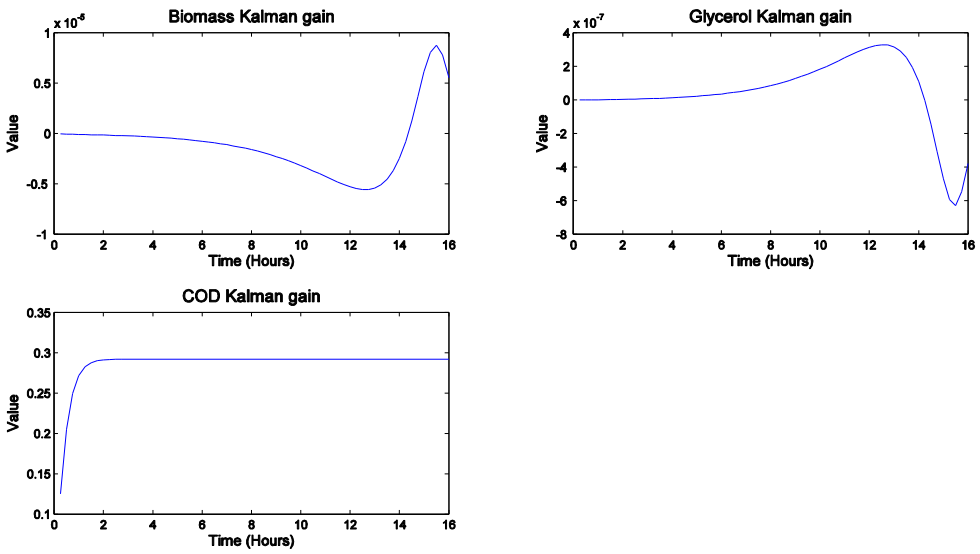

Figure 4-16. UKF Gains for exp15.

The Kalman filter gains, as observed in Figure 4-16, for *X* and *S* are variant towards the end but tend to return to zero and stabilize the system. While for *Cod* reaches a constant value. All of this probes the filter is stable for the validation experiment.

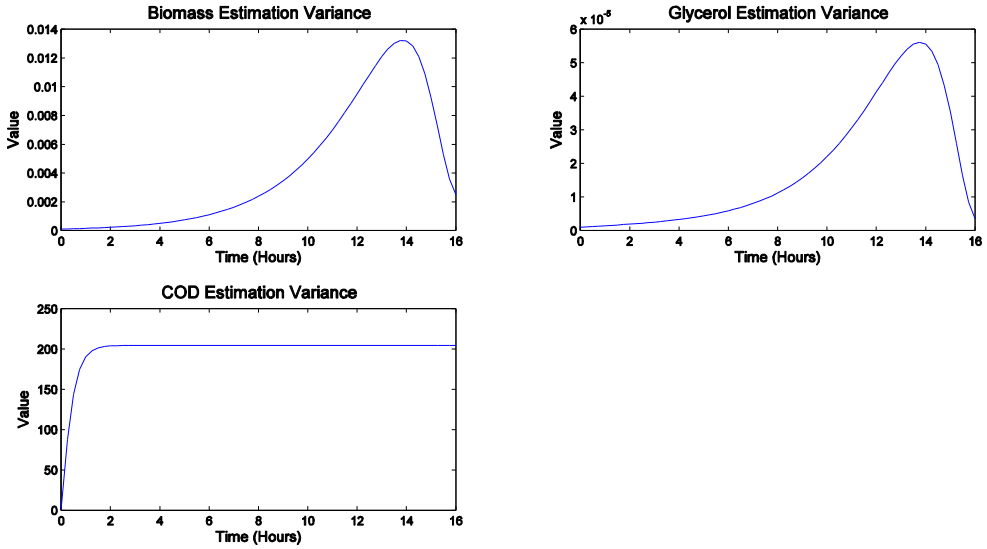

Figure 4-17. UKF Validation variances.

Figure 4-17 illustrates that the estimation of the states of interest *(X and S)* is done properly because their variances are small and most importantly due to the fact that both tend to converge to zero. Recurrently, *Cod* converges to a constant value as the other experiments.

4.3.2. Substrate Oxygen Saturation Model UKF

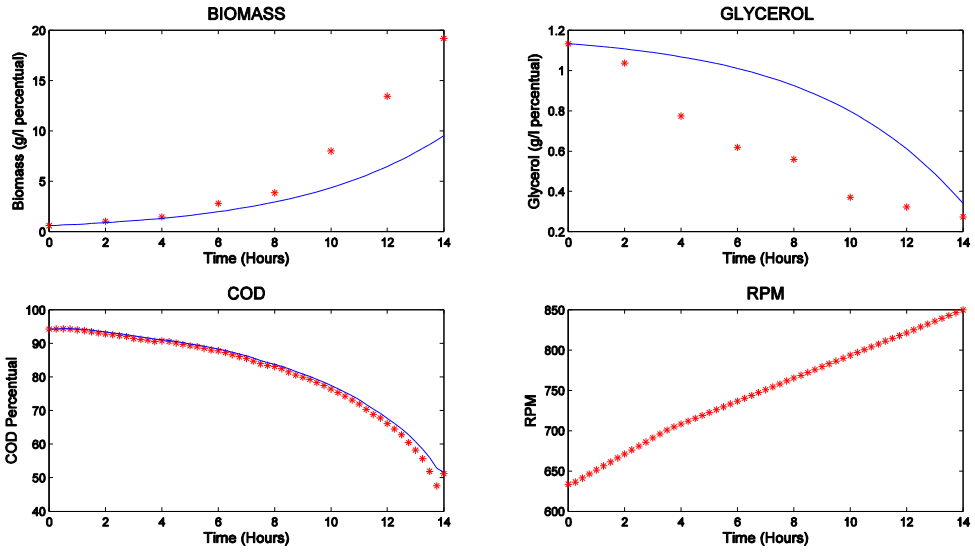

Figure 4-18. UKF exp11 Saturated Monod Model.

Figure 4-18 displays similar results as the UKF for the Standard Monod Model, where *X* and *S* are under and over the data with a noticeable error, failing to estimate correctly this states

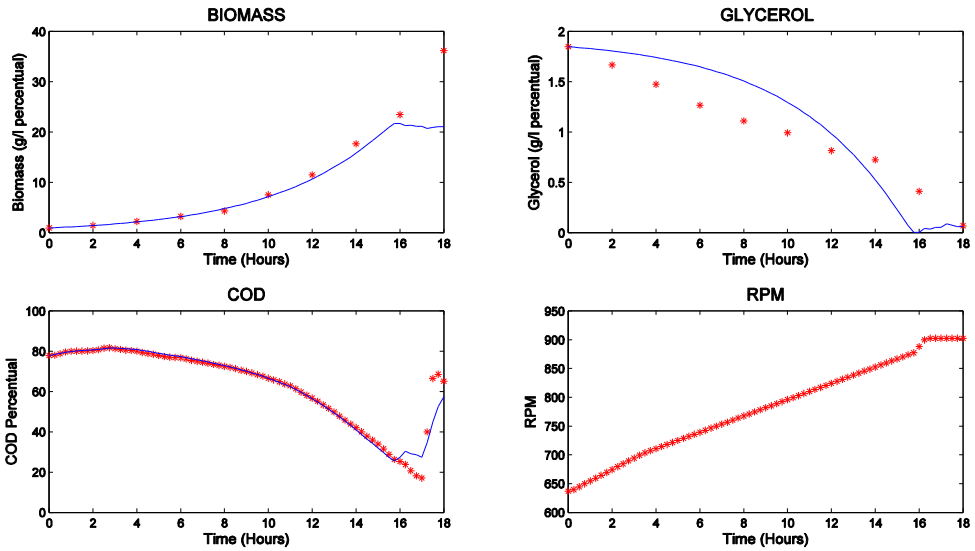

Figure 4-19. UKF exp21 for Saturated Model.

Figure 4-19 presents that just as with the EKF this model consequently has an end time saturation and ripple for *X* and *S* states. Increasing the estimation error of the model during the last two hours.

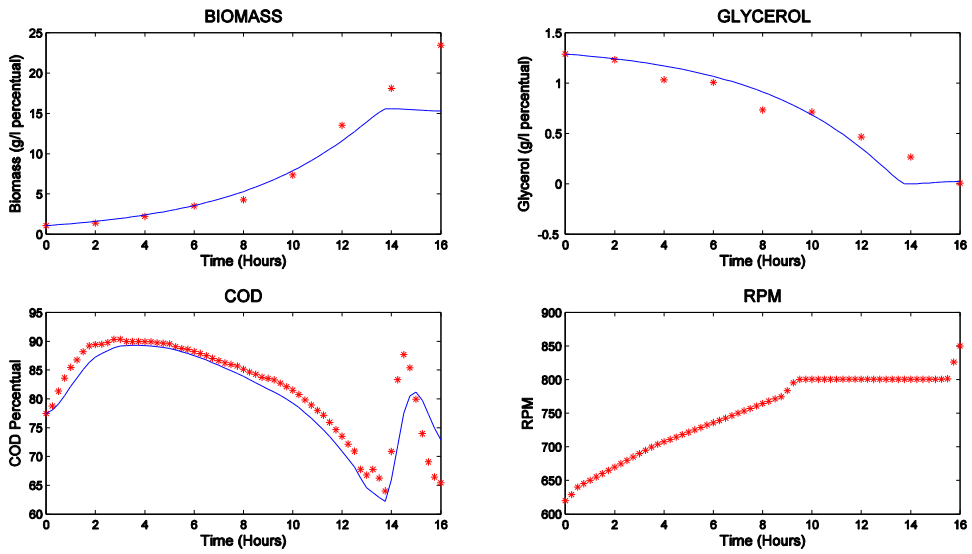

Figure 4-20. UKF Validation for Saturated Model with exp15.

Regarding the filter's validation, as can be seen in Figure 4-20, the *Biomass* and Glycerol estimations saturates towards the end, from 12th to 16th hour. Also, during this interval the signal displays smoothness, distinct to the ripples shown in the EKF for *exp21*. Moreover, *Biomass* state doesn't achieve its final value ensuing in an offset error.

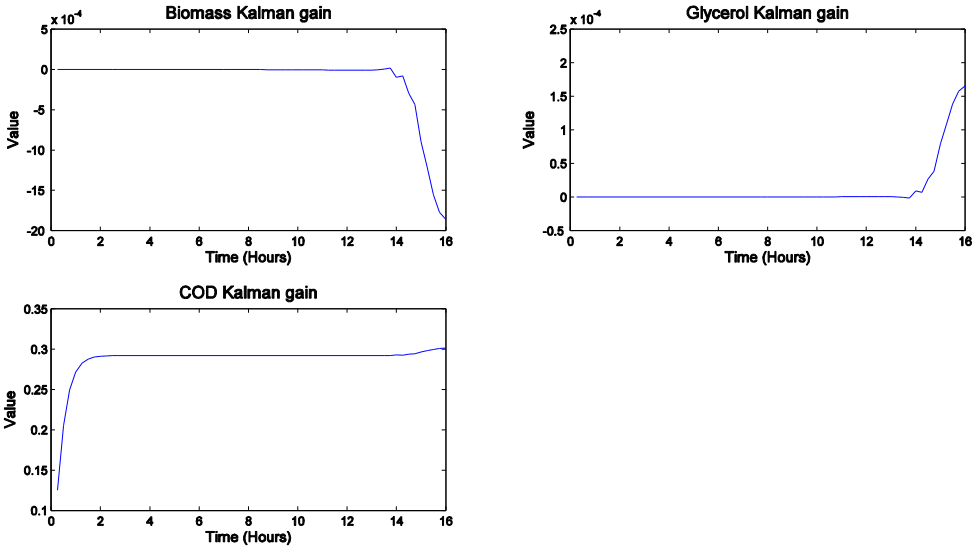

Figure 4-21. UKF Saturated Model Kalman gains.

Figure 4-21 show how unlike the EKF, the UKF gains for this particular model aren't variant over the time but tend to diverge at the last two hours of the simulation. Maintaining steadiness for closely the first twelve hours of the experiment.

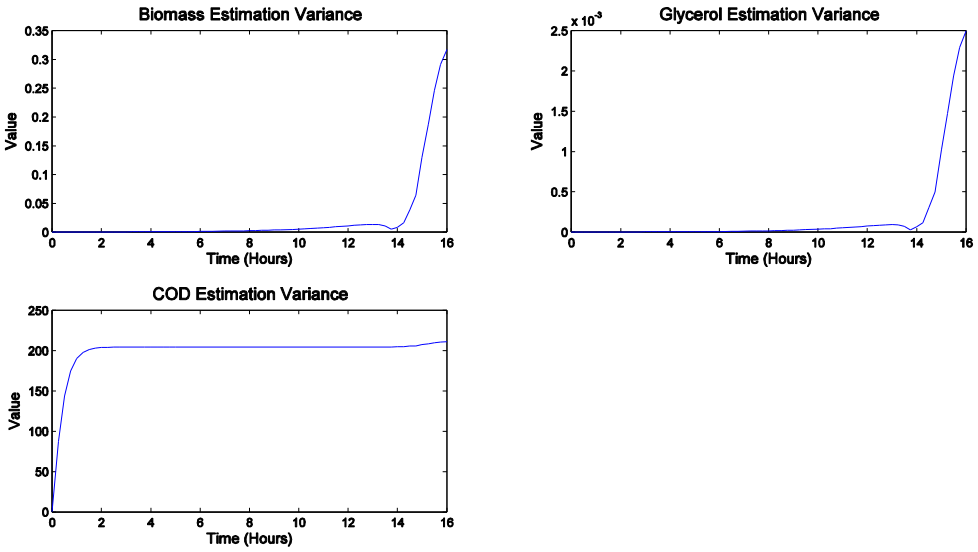

Figure 4-22. UKF Saturation Model states variances.

Concerning the estimation variances for the UKF Saturated Model observed in Figure 4-22, State variance for *Cod* is stable, while *X* and *S* does not. The first clinches a constant value, while *Biomass* and *Glycerol* variances are divergent starting from the 14th hour to the end of the experiment, exposing how the estimation for both states during this interval and afterwards is uncertain.

States	*Standard Monod UKF*		*Substrate Saturated UKF*	
	RMSE	*Var*	*RMSE*	*Var*
$\boldsymbol{X}$ $[\boldsymbol{g/l}]$	1.755	2.887	2.8603	8.1816
$\boldsymbol{S}$ $[\boldsymbol{g/l}]$	0.119	0.0081	0.1319	0.0174
$\boldsymbol{Cod}$ [%]	3.2576	8.9141	3.0170	9.1023

Table 15. UKF's RMS error and Variances for each state.

Table 15 resumes the RMS error and variance of Unscented Kalman Filters for both Models (Standard Monod & Monod Saturated Model). Standard Monod Model has better estimation performance for the unmeasured States (*X & S*) than the Saturated Model. For this reason, UKF with Standard Monod model is the preferred filter from this section.

4.4. Soft Sensor

A soft or virtual sensor is a kind of estimator, where the objective is exactly the same of the Kalman filter process; minimize the error between real and predicted states. The structure is a predefined or standard known structure, this architecture of the sensor is configurable; which is part of the design of the estimator; setting up how many parameters are there to be found. Correspondingly, since the structure is generalized and not a model it can be applied to a great variety of problems, nonetheless the parameters identified here are for this problem alone and moreover doesn't have any physical, chemical or biological meaning. Given that a bioprocess involves a living organism, which is a complex process by itself, plus all the variables, conditions, factors, disturbances, etc. makes these systems sometimes hard or tedious to model. Therefore, a soft/virtual Sensor is an attractive approach for these processes, mostly because it condenses and simplifies all of these dynamics and uncertainties in a generic model, as a consequence of the latter these sensors are known as *Black Box Models*.

Regarding what soft sensor is utilized the training and operation process is the same for anyone. The task is done by taking a *Regression Matrix* run its data with the respective operations and algorithm steps and return the corresponding prediction as shown in Figure 4-23. The regression Matrix $\varphi(t)$ consists of a set of inputs, and past outputs (in some architectures).

Figure 4-23. Soft/Virtual Sensor Estimator.

Once more the problem of the two sampling frequencies presents a challenge, since it's not just as simple as providing the inputs and desired output vectors to set up the sensor's training. The issue is resolved by arranging the regression's matrix elements so that each column matches correctly the desired output value. Recalling that regression elements for *Biomass* estimation are Cod and $\hat{X}$ present and/or past values. This can be observed in Figure 4-24.

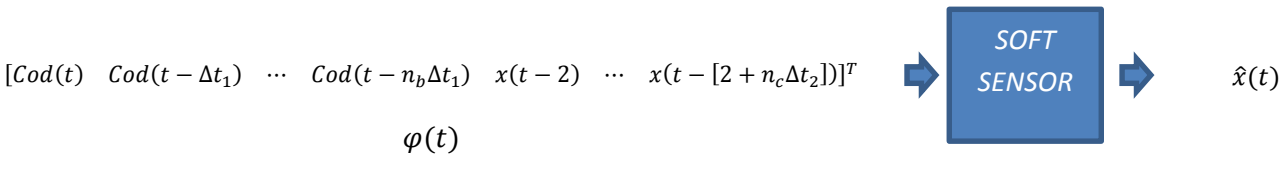

Figure 4-24. Regression Matrix for soft sensor

The latter is similar for doing the *Glycerol's* estimation as well.

The soft sensor alternatives used in bioprocesses applications (12) (13) are: Polynomial functions, Neural Networks, fuzzy systems, etc. As we mentioned before the sensor's architecture is customizable, hence its complexity growths with the order selected and proportional to the number of inputs taken to build the regression for each estimate, it impacts in the number of parameters to be found for the sensor system. Using the regression matrix in Figure 4-24 the number of inputs are $n_b + n_c$; where $[n_b: \#\ past\ values\ of\ Cod\ , n_c: \#\ Past\ values\ of\ \widehat{X}]$ .For example if we used a third order polynomial regressor the number of parameters of the sensor are $3 * (n_b + n_c) + 1$, that for the problem in question even though is a high number for the small data set acquired and may lead to *over fitting*. While the Neural Networks *(NN's)* with the same data inputs and order require $3 * (n_b + n_c) + 3 + 4$ parameters, even though there are six more, NN's offer more complex architecture than just simple equation's calculation like Polynomial functions. Therefore, NN's is the preferred soft sensor for use in this project.

The architecture of the Neural Network is presented in +++++ , as can be observed the Neural network doesn't utilize the input data *(Agitation -* $rpm(t)$*)* it only requires the initial conditions of the variable Dissolved Oxygen Concentration ($Cod(0)$) and its continuous measurements during the run of the experiment. Also the major difference with the model based estimator is that the neural network can only provide the estimation of one variable either *Biomass or Glycerol*, to achieve the estimation of the other variable it's necessary to train and implement another neural network for this state, which was done for consistency and comparison performance purposes against the Kalman Filter, in this case for the Substrate *Glycerol*.

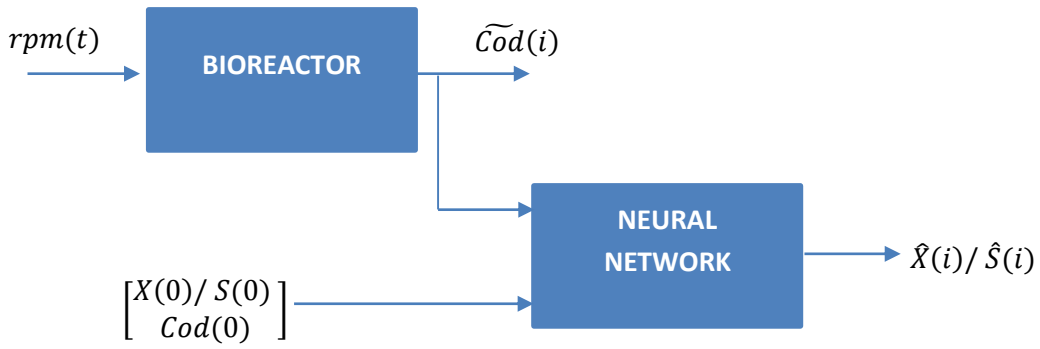

Figure 4-25. Neural Network Estimator block diagram.

4.4.1. Neural Networks

A Neural Network (NN) is an array of units named *PERCEPTRONS*. These entities provide a single value after the weighed summation of its inputs and the bias term passed through an activation function (either discrete or continuous), which puts the final mathematical operation to the sum and result of the perceptron. The perceptrons are arranged in interconnected layers, where the outputs of the previous layer are inputs for the next one and so on until reaches the final or output layer. The input layer (1st layer) is the one that operates the regression vector $\varphi(t)$ directly, the other(s) in between these two compose the hidden layer(s).

As was explained before it's just an introduction of what a NN is. Since the process at hand *(Bioreactor)* is nonlinear it is necessary to use a Nonlinear Neural Network, which is explained in the next section.

4.4.2. Nonlinear Neural Networks (NNN's)

The Nonlinear Neural Network has the same structure and basic components as explained before, the difference lies in the *activation function*, the most used ones are: *identity, Linear function, binary or sigmoidal function*. For this network the chosen function is the *sigmoidal* or *tansig* employed for the hidden layers, which allows the net to appropriate nonlinear characteristics and approximate these types of problems. As observed in the net example showed in Figure 4-26, the net has only one hidden layer which is also the input layer, and this one has the nonlinear activation function *(tansig)* for all 10 perceptrons or neurons and just one output neuron which in all cases is linear.

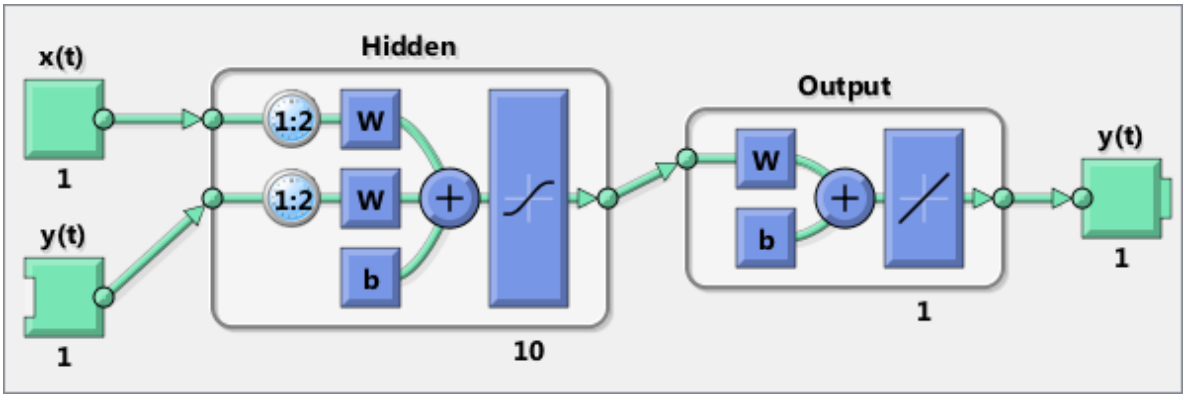

Figure 4-26. Nonlinear Neural Network example (NARX).

Therefore, w & b are the parameters for each layer of the NN to be identified and adjusted accordingly to approximate the output data set(X *or* S).

Given the characteristics of the problem at hand, the best approach for the NN's is to set up the regression matrix $\varphi(t)$ with measurements and past values of the state being estimated, $Cod(t)$ *and* $X(t)/\ S(t)$, thus benefit to employ the maximum information available for the estimation. The neural networks architectures that allow the aforementioned are: *Nonlinear AutoRegressive eXogenous input (NARX)* and *Nonlinear Output Error (NOE).* The NARX uses the measurements of both $Cod(t)$ *and* $X(t)$ as inputs; for *Biomass* estimation case; whereas NOE net utilizes the same $Cod(t)$ values but previous estimates $\hat{X}(t)$ instead of the real values, producing a closed loop Neural Network.

The search process for the NN's soft sensor is identifying the NARX (Figure 4-26) net parameters first and then using those parameters to simulate the NOE. Eventually the estimator must operate with just the Cod measurements and the initial conditions of each variable, just like the Kalman Filter. The resulting net estimates $X(t)/\ S(t)$ presents:

- Fifteen minutes (15') estimates due that is the best regression matrix $\varphi(t)$ that maximizes the existing data.
- Each estimate is composed of the array $Cod(t)$ values from the present time through the last two hours and also the *Biomass/Glycerol's* value from two hours before ($X(t-2)/\ S(t-2)$).
- The latter is product of the different sampling frequencies between variables; for Cod is a fifteen minutes sampling period while for *Biomass/Glycerol's* period is of two hours; setting up the estimator with two hour value discrepancy from the training step.

An example of the explained training set up for a NARX net with *exp11* is shown in Figure 4-27.

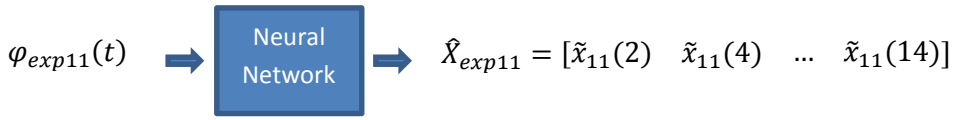

Figure 4-27. Network training setup with exp11.

Where the regression matrix is:

$$\varphi_{exp11}(t) = \begin{bmatrix} Cod(2) & Cod(1.75) & Cod(1.5) & Cod(1.25) & Cod(1) & Cod(0.75) & Cod(0.5) & Cod(0.25) & X(0) \\ Cod(4) & Cod(3.75) & Cod(3.5) & Cod(3.25) & Cod(3) & Cod(2.75) & Cod(2.5) & Cod(2.25) & X(2) \\ & & \vdots & & \vdots & & \vdots & & \\ Cod(14) & Cod(13.75) & Cod(13.5) & Cod(13.25) & Cod(13) & Cod(12.75) & Cod(12.5) & Cod(12.25) & X(12) \end{bmatrix}^T$$

The Networks were trained and simulated using Neural Networks toolbox of Matlab, normally it only requires providing the networks parameter architecture (number of neurons, past inputs, past outputs and network type) and the corresponding input and output vectors. Nonetheless, because of the two different sampling frequencies, the implementation was done as simple *Feedforward Network* (Figure 4-28) for the reason that is only net with the freedom to previously arrange the input matrix and set the NN for the corresponding regression matrix and net architecture according to the user, which is what we need in this particular case.

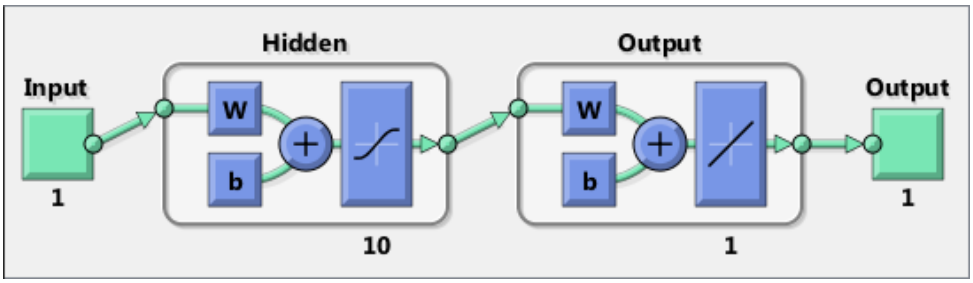

Figure 4-28. Feedforward Neural Network.

Since the regression matrix is customizable, it can be also exploited to identify the net as a multiple set training, same as in section 3 with *exp11 & exp21*. The multiple experiment regression matrix and desired outputs is denoted as observed in Figure 4-29. This consists of concatenating by column the corresponding regressor of each experiment and desired output achieving a single set of parameter for the net, leaving *exp15* again as the validation experiment.

$$\varphi(t) = \begin{bmatrix} \varphi_{exp11}(t)^T \\ \varphi_{exp21}(t)^T \end{bmatrix} \Rightarrow \boxed{\text{Neural Network}} \Rightarrow \hat{X} = \begin{bmatrix} \tilde{X}_{11}^T(t) \\ \tilde{X}_{21}^T(t) \end{bmatrix}$$

Figure 4-29. Multiple experiments Training setup.

Subsequently with the net parameters estimated, the following step is simulating the net as the 15' estimator. To achieve this, is necessary to create an initial vector of values for each variable (X, S, Cod) that goes backwards through to $t = -1.75$ $(hours)$. Each vector is created with the repeated value of its initial condition $(t = 0)$ seven times for the corresponding variable. All of this to provide the required data values for the proper estimation from $t = 0.25$. The $\varphi(t)$ regressor is extended for each experiment by seven new columns at the beginning of the matrix.

Once been described the soft sensor their results are presented next. The optimal Neural Network setup is with three hidden layer nonlinear neurons, eight $Cod(t)$ measurements (the current evaluation time plus seven immediately previous ones) and the delayed value of $X(t)/\ S(t)$ from two hours before.

4.4.3. Results and Validation

In this section are displayed the simulation results for the search process of an adequate Nonlinear Neural Network estimator for both *Biomass* and *Glycerol*, although the project's objective is only the former for matters of consistency and comparison results with the model based estimator the latter is estimated as well.

The method consisted on first, training the NN as *NARX* net using the desired outputs (*X or S)* also as part of the inputs *(measured values of the estimated variable)* estimating on a two hour period basis. Second, reconfigure the net as an Output Error net *(NOE)* where now the previous values of the output variable are the estimated ones, so that the NN requires only the measurements of *Cod* and the initial value of the assessed variable. Finally, extra labor is done to configure the net as a fifteen minute estimator to work similarly to the Kalman Filters.

The simulations of the first step are presented for each variable, contrary to the Kalman filter where they were displayed by experiment, the results are:

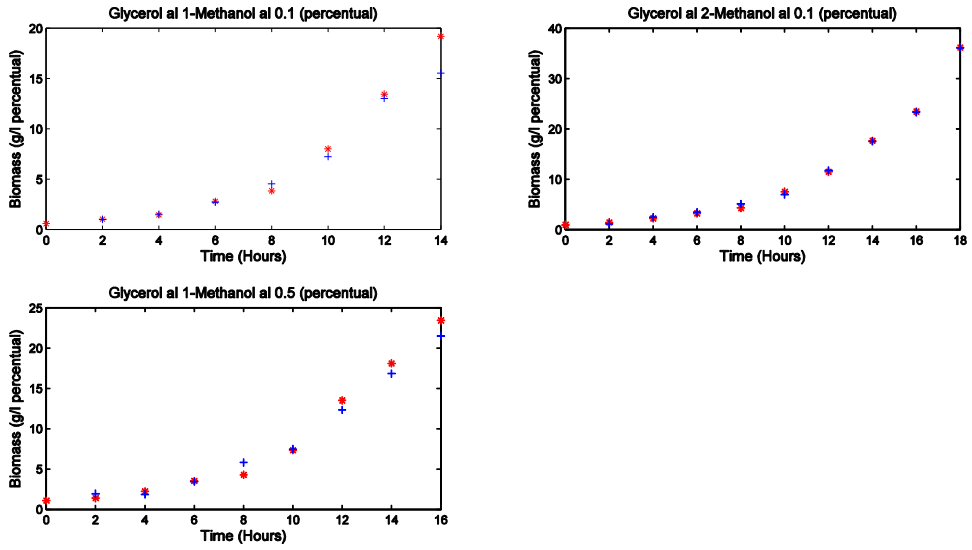

Figure 4-30. Neural Network Biomass estimation (Open loop).

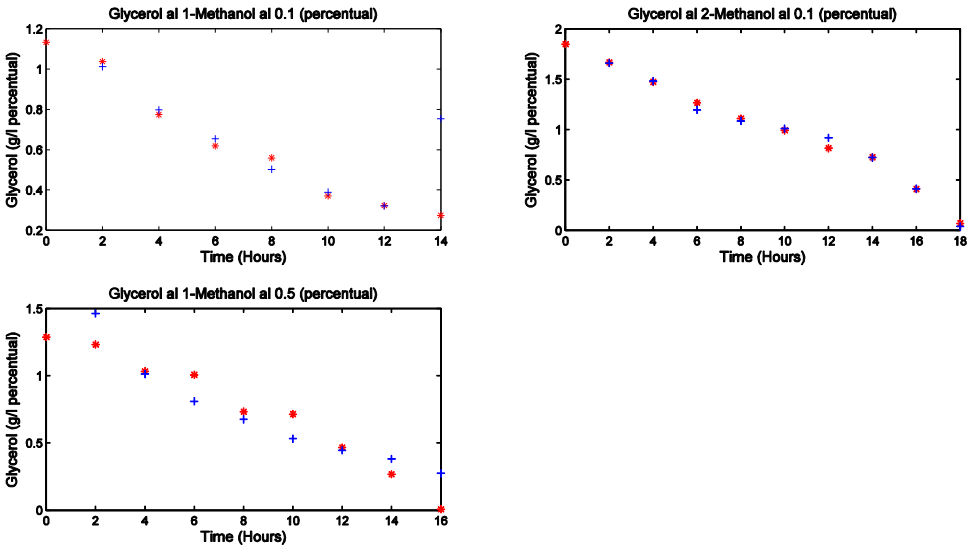

Figure 4-31. Neural Network Glycerol estimation (Open loop).

Figure 4-30 & Figure 4-31 shows the training results for variables *X* and *S*, respectively. The net was capable of fitting both the *Biomass & Glycerol* estimations for both training experiments; *exp11 & exp21 (top of the figure)*; and validation *(exp15-bottom)*. The only perceivable values outside the expected are: *the lasts Biomass & Glycerol values for exp11 and first estimation of Glycerol for exp15.*

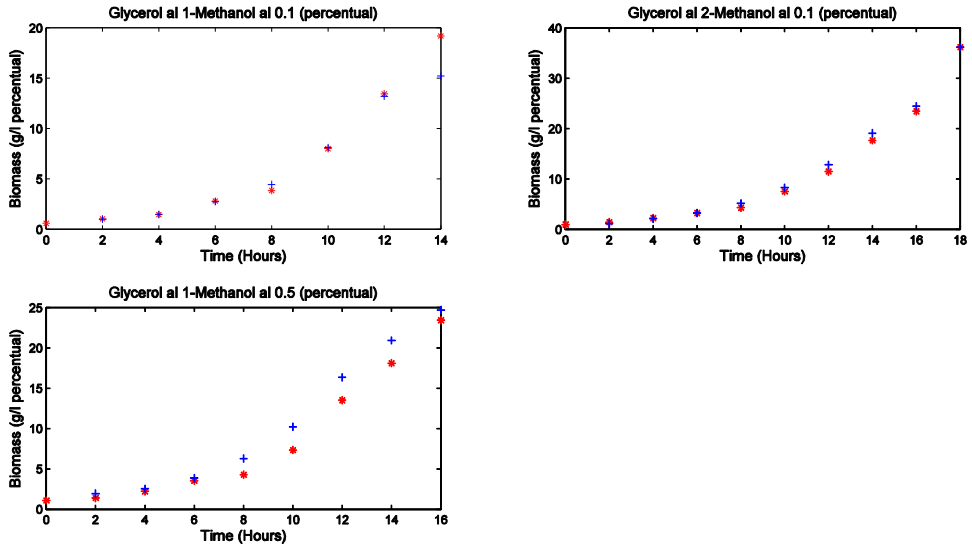

Figure 4-32. Neural Network output error Biomass estimation (Closed loop).

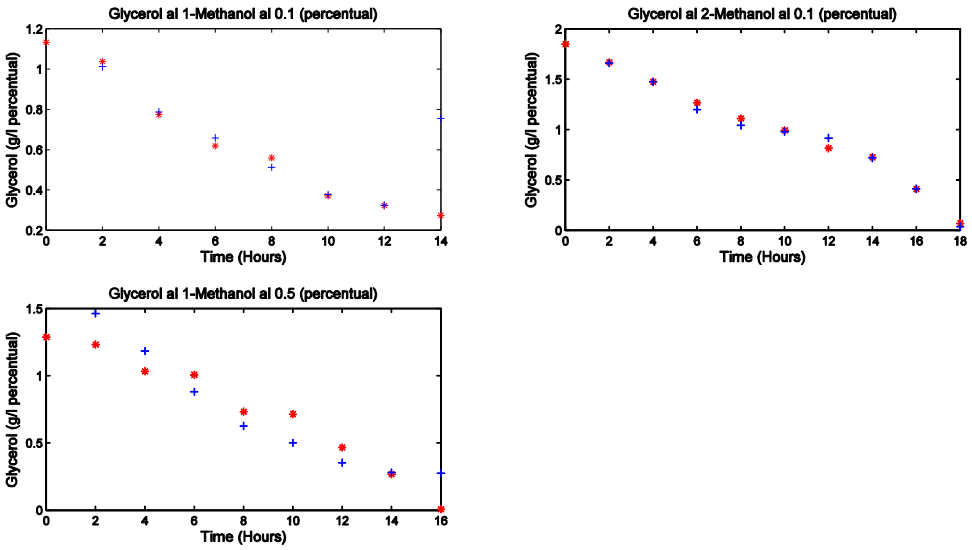

Figure 4-33. Neural Network output error Glycerol estimation (Closed loop).

Figure 4-32 & Figure 4-33 presents the closed loop NN simulation. The results are very similar to the open loop dynamics. The experiments maintain the variable progression and even the same odd values as before *(exp11 14th hour X & S values and final point of exp15)*. Ratifying the soft sensor's consistency after changing its structure with the feedback of the estimation and not the measured value. We can affirm virtual sensor's successful performance as an estimator. Nevertheless, the estimations can only be obtained on a two hour period basis, so we proceed with the results of the fifteen minutes estimator.

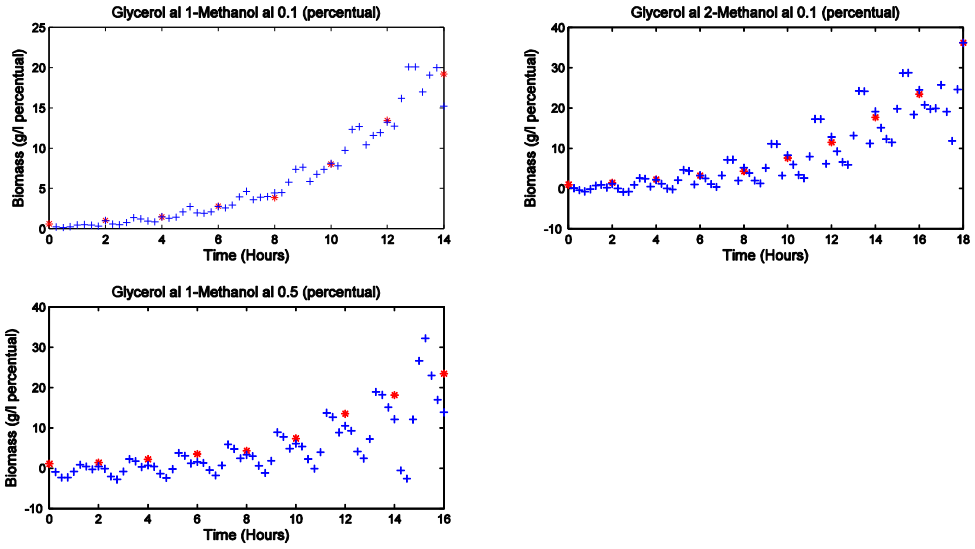

Figure 4-34. NN 15 minute Biomass estimator (Closed loop).

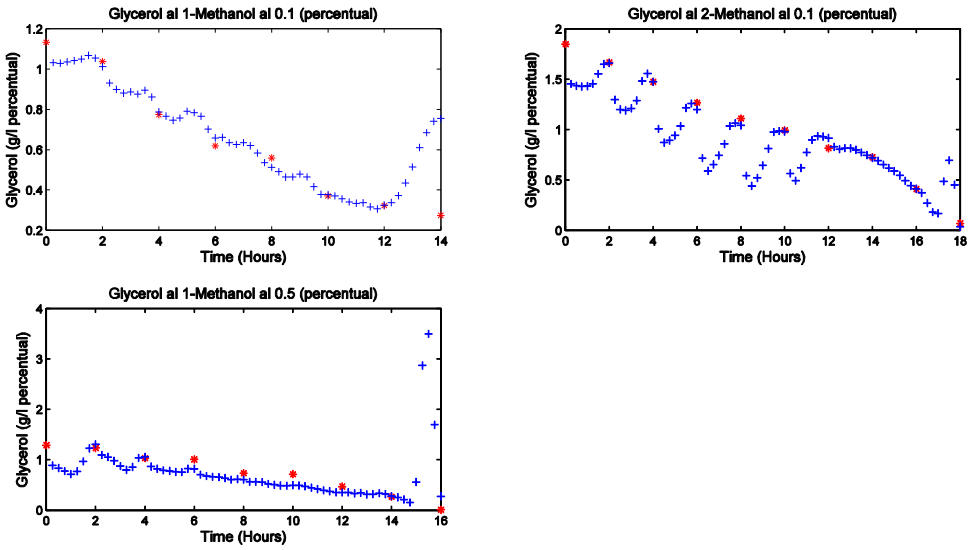

Figure 4-35. NN 15 minute Glycerol estimator (Closed loop).

Figure 4-34 & Figure 4-35 shows how the seven intermediate ($e.g., t = 0.25, 0.5, \ldots, 1.75\ [hours]$) samples from the two hour intervals doesn't present consistent dynamics *(divergence, instability, transients, off trends etc.)* through the different experiments, not even during the training sets. Therefore, this estimator is determined as unreliable given it cannot assure the correct estimation of these time points. As described by *Biomass* estimates from *exp21 & exp15* and *also* from *Glycerol's exp21*. Asserting that the 15 minute estimator is not a suitable solution for this process.

The validation experiment estimation Root Mean Squared Error and its variance is showed in Table 16:

$\boldsymbol{Variable}$	$\boldsymbol{NN-NARX\ (Open\ Loop)}$		$\boldsymbol{NN-NOE\ (Closed\ Loop)}$	
	$RMSE$	Var	$RMSE$	Var
$\boldsymbol{X}\ [\boldsymbol{g/l}]$	1.09	1.238	1.952	1.305
$\boldsymbol{S}\ [\boldsymbol{g/l}]$	0.163	0.0302	0.171	0.033

Table 16. RMS Error & Variance for validation exp15 (Open-Close loop)

As expected the Neural Networks performance for each variable is acceptable. Neither performance parameter is affected much after the net changes from being a *NARX* to *NOE*. The estimations of the soft sensor are remarkable compared to the max values of each variable displaying high accuracy and consistency. However, the net only provides the estimations on a two hour period basis.

Now that the Soft sensor has been validated, it is necessary to select which estimator will be implemented on Labview as explained in the next section.

4.5. Estimator Selection

Since our interest is *Biomass* estimation, the estimator is selected between the best filter from the model based Kalman section; recalling that the best model based estimator is the Extended Kalman Filter with standard Monod model; and the Neural Networks presented before. We begin by comparing the same performance features used during their validation: *Root Mean Squared Error and Variance*. This is described in Table 17:

$\boldsymbol{Variable}$	$\boldsymbol{Standard\ Monod\ EKF}$		$\boldsymbol{NN-NOE\ (Closed\ Loop)}$	
	$RMSE$	Var	$RMSE$	Var
$\boldsymbol{X}\ [\boldsymbol{g/l}]$	1.755	2.885	1.952	1.305
$\boldsymbol{S}\ [\boldsymbol{g/l}]$	0.119	0.008	0.171	0.033

Table 17. EKF-NN performance comparison.

The results are conclusive, first *Standard Monod EKF (SM-EKF)* surpasses the *Neural Network* in both performance characteristics by a slight margin for *Biomass' and Glycerol's* estimations. Also regarding the quality features of each estimator, *SM-EKF* is considerably better than *NN* at describing the process evolution during time with consistency and softness over all the course of the experiment, not only for the time values where the data was sampled for training purposes. The prior is a highly valued characteristic for the estimator for the reason that the objective is to be aware of the *Biomass* value and dynamics during the course of the growth phase in order to assert how this stage is going and confirm its correct development to the point where the substrate can be changed to *Methanol* and proceed to protein production phase at the right time.

For all the explained before the selected estimator for implementation is the *Standard Monod Extended Kalman Filter*.

5. LABVIEW IMPLEMENTATION

Remembering that the process is already automated via Labview system, the procedure was adding the *Standard Monod Extended Kalman Filter (SM-EKF)* to the existing architecture. The goal is to provide the real time value of the Biomass in an additional graph to the already existing ones (Cod, RPM, temperature, pH, etc.), so that the supervision of such important variable is at hand too and allows a better decision making to correct the process and manage the experiment on a successful course.

5.1. Estimator Development

The operation of the *SM-EKF* demands the data values of *Dissolved Oxygen Concentration (Cod)*, agitation *(rpm)* and initial conditions of the three states *(Biomass-X, Glycerol-S and Cod)* to start running. The procedure to integrate it into the Labview environment starts by acquiring the present *Cod & rpm* values, coming from the online feed of the virtual sensors inside the program and storing it in a vector. Then, calculating the average of this fully stored vectors to obtain the 15 minute *Cod & rpm* states and supplying it to the *SM-EKF* to process it to produce the estimate at the current time. The last is showed in Figure 5-1:

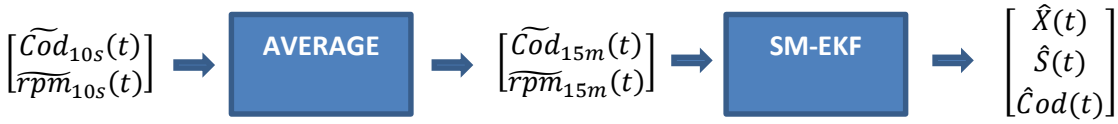

Figure 5-1. Estimator Implementation process.

The *SM-EKF* is implemented through the *Mathscript* programming node *(MS-N)*. Which allows to maintain the same programming code used during Matlab simulations, since *MS-N* accepts .m files as functions or entire scripts, but with restricted to the use of small database basic functions. To utilize a full Matlab environment it is necessary to have a Full and professional development system Labview to use the Matlab script node instead of the *MS-N*. The Matlab script node works by executing the script directly in Matlab environment with all its functions/toolboxes and not in the smaller compiler integrated to Labview for this particular programming language. Since the *SM-EKF* is composed by simple mathematic operations, nonlinear function evaluation and linear algebra for matrixes, the filter can be implemented using the *MS-N.*

The estimator implementation as told before only requires the initial condition of the states *(X, S & Cod)* and the online measurement of *Cod*. Therefore we set up the interface to ask for these initial values and display the estimated Value of *Biomass*. This is shown in Figure 5-2:

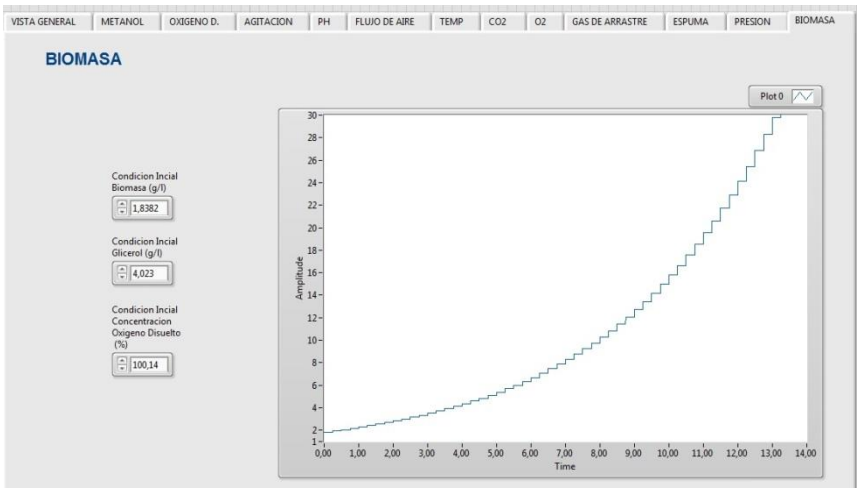

Figure 5-2. Biomass Estimator Interface.

5.2. Results

The implementation phase was validated with a new data set different from the ones used before in sections 3 & 4, this experiment initial condition was:$[X(0), S(0), Cod(0)]^T = [1.83, 4.1, 100.4]^T$, With a duration of 12 hours and its results are displayed in Figure 5-3. Clearly, the estimator follows the *Biomass* dynamic but over the measured values. Its performance characteristics are $RMSE = 3.304(g/l)$ & $Var = 3.822$, that compared to the simulation stage are higher, but still are within an acceptable range. The validation is recognized as satisfactory acknowledging the correct estimator implementation.

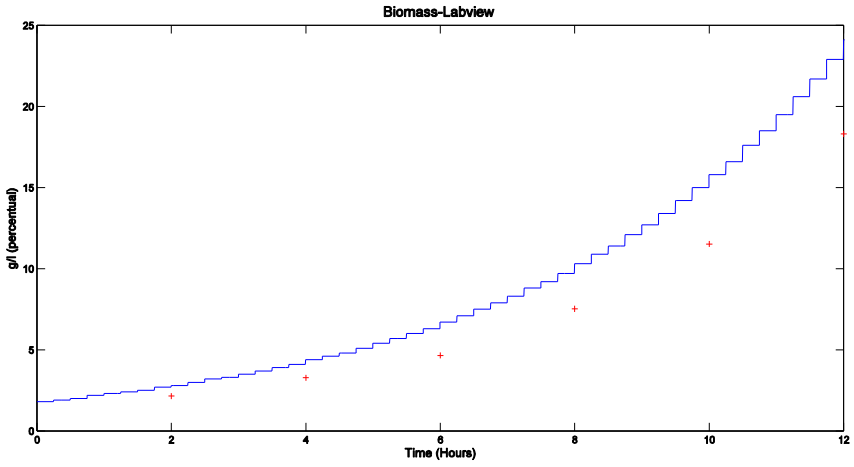

Figure 5-3. Labview Validation Results.

6. CONCLUSIONS

Using a three experiment set we were able to acquire a successful differential equation model and Neural Network *(NN)* that adjusted correctly to the data set to acceptable values by minimizing the estimation error on each design. The latter even though that the data set is small regarding the amount of total data points as consequence the two hour sampling period for *Biomass*/*Glycerol* and the number of approved experiments. The model had positive effectiveness as it was used in the Nonlinear Kalman Filter estimators successfully, only requiring the measurements of *Cod & rpm* plus the initial conditions of the estimated states *(X, S & Cod).* The net estimated satisfactorily the states for the sampled time points using the same architecture, which is known as *Nonlinear Output Error* an closed loop net.

The estimator selected for implementation in Labview environment was the *Extended Kalman Filter with Standard Monod Model*. Because it is the estimator with best performances clearly surpassing the NN both quantitatively and qualitatively, offering the lowest RMS error and variance for both *X & S* and also the highest consistency, stability and smoothness for the dynamics over the time. Its functioning was validated with a new data set over Labview obtaining satisfactory results regarding *Biomass* dynamics, *RMSE* and error *Variance*, corroborating that is possible to implement complex estimators as *Kalman Filters* in its environment and possibly the *Neural Networks* as well.

Despite that the data set had two different sampling frequencies; two hour period for *X & S* and ten second for *Cod & rpm;* the training and validation procedure for the estimators was done correctly. For the model based was done by modifying the objective function weighing cost as a time variant matrix, as for the Neural Network was avoided by arranging the input regression matrix to do the corresponding estimations by matching its inputs for the respective time. By having a customizable regression matrix as further work a fifteen minute net was implemented which tried to follow the system dynamics, however wasn't able to predict correctly these intermediate time values.

In the Bioreactor Modeling stage we were able to confirm that for the growth phase the best model for *Pichia Pastoris* yeast is the *Monod Model* based on Monod's growth rate for microorganisms, contrary to what was achieved in the production phase for the same yeast. The *Saturated Monod Model* for the Kalman Filtering wasn't able to produce reliable estimations, since its Kalman gains are instable and variances are divergent at the end of the simulations.

After reviewing the results in section 3.4 regarding the different combinations of data sets for training and validation. It can be noted that the estimator is not highly consistent for all the sets given the RMSE and variance differ through the sets, and these does not seem to belong to the same statistical distribution.

1.2 Future Work

The methodology and work developed in this project can be applied to the different strains used at IEIM for the *Pichia Pastoris* yeast. Therefore, for all the process developed at the institute a better grasp and knowledge of the bioreactor dynamics and evolution during the growth phase is obtained.

Improve and refine more the current model and filter on the basis that with a larger acquired data set, given that this new information their performance is benefited.

Apply the Extended Kalman Filter for the production phase of the yeast, so that it can be exploited the *Biomass'* estimation for a closed loop control for this stage and improve the production of the protein, since for feedback control systems the values of all the variables is required for the control system operation.

7. REFERENCES

1. **Barragan, D. Martinez.** *Optimizacion de la producción de proteínas recombinantes en Pichia basada en un modelo in silico.* Pontifica Universidad Javeriana. Bogotá : s.n., 2014. Trabajo de Grado de Maestría.

2. **A., Ballesteros B. Mayra.** *SISTEMA DE MONITOREO DE VARIABLES EN UN BIORREACTOR PARA LA.* Pontifica Universidad Javeriana. Bogotá : s.n., 2014. Trabajo de Grado de Maestría.

3. *A strutctured kinetic model for recombinant protein production by Mut+strain of Pichia Pastoris".* **E. Çelik, P. Çalık, and S. G. Oliver.** 2009, Chem eng. Sci., Vol. 64, págs. 5028-5035.

4. *Dynamic modeling of methylotrophic Pchia pastoris culture with exhaust gas analysis: From cellular metabolism to process simulation.* **H. Niu, M. Daukandt, C. Rodriguez, P. Fickers, and P. Bogaerts.** 2012, Chem. Eng. Sci., Vol. 87, págs. 381-392.

5. *Development of a mathematica model for the growth associated Polyhydroxybutyrate fermentation by Azohydromonas australica and its use for the design of fed-batch cultivation strategies.* **Gahlawat G., Srivastava A. K.** 2012, Bioresource Technology, Vol. 137, págs. 98-105.

6. *Macrokinetic model for methylotrophic Pichia Pastoris based on stoichiometric balance.* **H. T. Ren, J. Q. Yuan, and K.-H. Bellgardt.** 1, 2003, J. Biotechnol., Vol. 106, págs. 53-68.

7. *Bioprocess engineering aspects of heterologous proten production in Pichia pastaris: A review.* **Potvin G., Ahmad A., Zhang Z.** 2010, Biochemical Engineering Journal, Vol. 64, págs. 91-105.

8. **B. Houska, D. Ariens, H.J. Ferreau.** ACADO Toolkit user's manual. [En línea] 2010-2011. www.acadotoolkit.org.

9. **A. Gelb, J. F. Kasper, R. A. Nash, C. F. Price, and A. A. Sutherland.** *Applied optimal estimation.*

10. **Wan E., van der Merwe R.** *The Unscented Kalman Filter for nonlinear estimation.* Oregon Graduate Institue of Science & Technology.

11. *Unscented Filtering and Nonlinear Estimation.* **Julier S., Uhlmann J.** 3, 2004, Proceedings of the IEEE, Vol. 92, págs. 401-422.

12. *Soft sensor in bioprocessing: a status report and recommendations.* **R. Luttmann, D. G. Bracewell, G. Cornelissen, K. V Gernaey, J. Glassey, V. C. Hass, C. Kaiser, C. Preusse, G. Striedner, and C.-F. Mandenius.** 8, 2012, Biotechnol. J., Vol. 7, págs. 1040-1048.

13. *Artificial Neural Network based Soft sensor for fermentation of Recombinant Pichia pastoris.* **Geethalakshmi S., Pappa N.** 2010, International Conference on advances in computer engineering, págs. 148-152.

14. *Estimation of recombinant protein production in Pichia Pastoris based on a contraint-based model.* **M. Tortajada, F. Llaneras, D. Ramon and J. Pico.** 6, 2012, J. Process Control, págs. 1139-1151.

15. *Data-driven Soft Sensors in the process industry.* **P. Kadlec, B. Gabrys, and S. Strandt.** 2009, Computer and Chemical Engineering, Vol. 33, págs. 795-814.

16. **Avila, Y. G. Grace.** *Estimacion de Biomasa en un bioreactor.* Pontifica Universidad Javeriana. Bogota : s.n., 2013. Tesis.

17. *Soft sensor assisted dynamic bioprocess control: Efficient tools for bioprocess development.* **P. Sagmeister, P. Wechselberger, M. Jazini, A. Meitz, T. Langemann, and C. Herwig.** 2013, Chem. Eng. Sci., Vol. 7, págs. 190-198.

18. *On-line Estimation in Fed-batch Fermentation Process Using State Space Model and Unscented Kalman FIlter.* **J. Wang, L. Zhao, and T. Yu.** 2, 2010, Chinese Chem. Eng., Vol. 18, págs. 258-264.

19. **F. F. Lewis, L. Xie, and D. Popa.** *Optimal and Robust Estimation: with and Introduction to Stochastic Control Theory.*

20. *Experimental validation of an extended Kalman Filter estimating acetate concentration in E. coli cultures.* **L. Dewasme, G. Goffaux, a.-L. Hantson, and a. Vande Wouwer.** 2, 2013, J. Process Control, Vol. 23, págs. 148-157.

21. *Direct Filtering: A new approach to optimal filter design for nonlinear Systems.* **C. Novara, F. Ruiz, M. Milanese.** 1, 2013, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, Vol. 58, págs. 86-99.

22. *From model-based to data driven design.* **M. Milanese, F. Ruiz, and M. Taragna.** 2013, págs. 273-285.